

LNCS 2957

Peter Langendoerfer
Mingyan Liu
Ibrahim Matta
Vassilis Tsaoussidis (Eds.)

Wired/Wireless Internet Communications

Second International Conference, WWIC 2004
Frankfurt (Oder), Germany, February 2004
Proceedings



Springer

Lecture Notes in Computer Science

2957

Edited by G. Goos, J. Hartmanis, and J. van Leeuwen

Springer

Berlin

Heidelberg

New York

Hong Kong

London

Milan

Paris

Tokyo

Peter Langendoerfer Mingyan Liu
Ibrahim Matta Vassilis Tsaoussidis (Eds.)

Wired/Wireless Internet Communications

Second International Conference, WWIC 2004
Frankfurt (Oder), Germany, February 4-6, 2004
Proceedings



Springer

Series Editors

Gerhard Goos, Karlsruhe University, Germany
Juris Hartmanis, Cornell University, NY, USA
Jan van Leeuwen, Utrecht University, The Netherlands

Volume Editors

Peter Langendoerfer
IHP
Im Technologiepark 25, 15236 Frankfurt (Oder), Germany
E-mail: langendoerfer@ihp-microelectronics.com

Mingyan Liu
University of Michigan
1301 Beal Ave, 4238 EECS, Ann Arbor, MI 48109-2122, USA
E-mail: mingyan@eecs.umich.edu

Ibrahim Matta
Boston University
College of Arts and Sciences, 111 Cummington Street, MCS-271, Boston, MA 02215, USA
E-mail: matta@cs.bu.edu

Vassilis Tsaoussidis
Demokritos University
Department of Electrical and Computer Engineering
12 Vas. Sofias Str., 671 00 Xanthi, Greece
E-mail: vtsaousi@ee.duth.gr
Cataloging-in-Publication Data applied for

A catalog record for this book is available from the Library of Congress.

Bibliographic information published by Die Deutsche Bibliothek
Die Deutsche Bibliothek lists this publication in the Deutsche Nationalbibliografie;
detailed bibliographic data is available in the Internet at <<http://dnb.ddb.de>>.

CR Subject Classification (1998): C.2, H.3, H.4, D.2, H.5.1, K.4.4

ISSN 0302-9743

ISBN 3-540-20954-9 Springer-Verlag Berlin Heidelberg New York

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

Springer-Verlag is a part of Springer Science+Business Media
springeronline.com

© Springer-Verlag Berlin Heidelberg 2004
Printed in Germany

Typesetting: Camera-ready by author, data conversion by PTP-Berlin, Protago-TeX-Production GmbH
Printed on acid-free paper SPIN: 10984819 06/3142 5 4 3 2 1 0

Preface

The International Conference on Wired/Wireless Internet Communications (WWIC) was held for the second time, following a successful start in 2002, in Las Vegas. The goal of the conference was to present high-quality results in the field, and to provide a framework for research collaboration through focused discussions that designated future research efforts and directions. The number and the quality of submissions indicate that we are well on the way to establishing WWIC as a major event in the field of wired/wireless internet communications.

We received around 60 competitive submissions from Europe, North America, the Middle East and the Far East. Each submission was reviewed by at least two experts, although the majority received three or more reviews. Based on this rigorous reviewing procedure, the International Program Committee selected 26 submissions for presentation and publication in the proceedings. Therefore, we should all expect the quality of a selective conference in this volume. We hope you will enjoy it.

The papers selected for presentation at WWIC 2004 were stimulating and of utmost interest. They were organized into eight sessions:

1. Protocol engineering and energy efficiency in wireless networks
2. Mobility management and mobile devices
3. Transport layer and congestion control
4. Architecture, implementation and experimentation
5. Network and protocol modeling
6. Wireless network scheduling and analysis
7. Multimedia distribution and group communication
8. Service discovery.

We would like to thank the authors for choosing WWIC 2004 to submit their results. We would also like to thank all the members of the Technical Program Committee, as well as the additional reviewers, for their effort to provide detailed and constructive reviews.

Peter Langendoerfer
Mingyan Liu
Ibrahim Matta
Vassilis Tsaoussidis

Executive Committee

General Co-chair	Vassilis Tsaoussidis (Demokritos University, Greece)
TPC Chair	Ibrahim Matta (Boston University, USA) Peter Langendoerfer (IHP Microelectronics, Germany) Mingyan Liu (University of Michigan, USA)

Program Committee

Farooq Anjum	Telcordia Technologies (USA)
Torsten Braun	University of Bern (Switzerland)
Xiuzhen Cheng	George Washington University (USA)
Mark Crovella	Boston University (USA)
Klaus David	University of Kassel (Germany)
Olaf Droegehorn	University of Kassel (Germany)
Yuguang Fang	University of Florida (USA)
Jennifer Hou	University of Illinois at Urbana-Champaign (USA)
Yevgeni Koucheryavy	Tampere University of Technology (Finland)
Rolf Kraemer	IHP Microelectronics (Germany)
Srikanth Krishnamurthy	University of California, Riverside (USA)
Adrian Lahanas	University of Cyprus (Cyprus)
Peter Langendoerfer	IHP Microelectronics (Germany)
Victor Leung	University of British Columbia (Canada)
Ben Liang	University of Toronto (Canada)
Mingyan Liu	University of Michigan (USA)
Qingchong Liu	University of Oakland (USA)
Henning Maass	Philips Research Laboratories (Aachen)
Petri Maehonen	University of Oulu (Finland)
Qusay Mahmoud	Guelph University (Canada)
Christian Maihofer	DaimlerChrysler (Germany)
Ibrahim Matta	Boston University (USA)
Ioanis Nikolaidis	University of Alberta (Canada)
Guevara Noubir	Northeastern University (USA)
Jianping Pan	Fujitsu (USA)
George Polyzos	AUEB (Greece)
Jochen Schiller	FU Berlin (Germany)
Sherman Shen	University of Waterloo (Canada)
Ioannis Stavrakakis	University of Athens (Greece)
Vassilis Tsaoussidis	Demokritos University (Greece)
Dapeng Oliver Wu	University of Florida (USA)
Chi Zhang	Florida International University (USA)
Yongguang Zhang	Hughes Research Labs (USA)
Martina Zitterbart	TU Karlsruhe (Germany)
Michele Zorzi	University of Ferrara (Italy)

Organizing Committee

Daniel Dietterle	IHP Microelectronics (Germany)
Jan Schaeffner	IHP Microelectronics (Germany)
Heike Wasgien	IHP Microelectronics (Germany)

Reviewers

Peter Langendoerfer	Henning Maass	Yiannos Mylonas
Farooq Anjum	Tim Leinmueller	Dr. Vasos Vassiliou
Xiuzhen Cheng	Michael Meier	Latha Kant
Mark Crovella	Marcin Michalak	Lin Cai
Klaus David	Marc Heissenb'ttel	Junaid Asim Khan
Jochen Schiller	Ruy de Oliveira	Fei Yu
Sherman Shen	Attila Weyland	Hossam Fattah
Ioannis Stavrakakis	Dhanant Subhadrabandhu	Dmitri Moltchanov
Vassilis Tsaoussidis	Markus Baumeister	Wang Jian
Dapeng Oliver	Javier Espina	Aaron So
Chi Zhang	Heribert Baldus	Ahmed Zahran
Yongguang Zhang	Klaus Weidenhaupt	Mahdi Lotfinezhad
Martina Zitterbart	Jikai Li	Stephen Drew
Michele Zorzi	Gaoxi Xiao	Enrique J.
Olaf Droegehorn	Xudong Wang	Chih-fan Hsin
Yuguang Fang	Ariton Xhafa	Joe Fikart
Yevgeni Koucheryavy	Xiaolei Guo	Jim Cavers
Rolf Kraemer	Kejie Lu	Prashant Krishnamurthy
Srikanth Krishnamurthy	Song Ci	Jakob Eriksson
Adrian Lahanas	Onur Altintas	John Jones
Victor Leung	Linda Xie	Zhenqiang Ye
Ben Liang	Wei Li	Gentian Jakllari
Mingyan Liu	Kai Xing	Farshid Agharebparast
Qingchong Liu	Min Ding	Eleni Stroulia
Petri Maehoenen	Andrew Thaeler	Pawel Gburzynski
Qusay Mahmoud	Hartmut Ritter	Lefteris Mamatas
Christian Maihoefer	Gunter Schaefer	Janne Riihijarvi
Ibrahim Matta	Francesca Cuomo	Marina Petrova
Ioanis Nikolaidis	Giuseppe Bianchi	Honghai Zhang
Guevara Noubir	Leonardo Badia	Chunyu Hu
Jianping Pan	Elena Pagani	Tan F. Wong
George Polyzos	Xuan Li	
Jennifer Hou	Donna Jin	
Torsten Braun	Qiang Ni	

Table of Contents

Session 1: Protocol Engineering and Energy Efficiency in Wireless Networks

Distributed MAC Protocol to Improve Energy and Channel Efficiency in MANET	1
<i>C.-Y. Liu, C.-H. Lin</i>	
CGGC: Cached Greedy Geocast	13
<i>C. Maihöfer, R. Eberhardt, E. Schoch</i>	
Design of Energy Efficient Wireless Networks Using Dynamic Data Type Refinement Methodology	26
<i>S. Mamagkakis, A. Mpartzas, G. Pouiklis, D. Atienza, F. Catthoor, D. Soudris, J.M. Mendias, A. Thanailakis</i>	
Context-Aware Group Communication in Mobile Ad-Hoc Networks	38
<i>D. Bottazzi, A. Corradi, R. Montanari</i>	

Session 2: Mobility Management and Mobile Devices

An Empirical Wi-Fi Based Location Mechanism for Urban Search and Rescue Operations	48
<i>R. Aldunate, M. Nussbaum, F. Pena-Mora</i>	
Mobility-Aware Rendezvous Point for Mobile Multicast Sources	62
<i>I. Romdhani, M. Kellil, H.-Y. Lach, A. Bouabdallah, H. Bettahar</i>	
An Address Configuration and Confirmation Scheme for Seamless Mobility Support in IPv6 Network	74
<i>S.-H. Hwang, Y.-H. Han, S.-G. Min, C.-S. Hwang</i>	

Session 3: Transport Layer and Congestion Control

TCP Optimization through FEC, ARQ, and Transmission Power Tradeoffs	87
<i>D. Barman, I. Matta, E. Altman, R. El Azouzi</i>	
Sliding Mode Congestion Control in Differentiated Service Communication Networks	99
<i>H. Ebrahimirad, M.J. Yazdanpanah</i>	
Application of Robust Fuzzy Adaptive Second-Order Sliding-Mode Control to Active Queue Management	109
<i>M. Jalili-Kharaajoo</i>	

Graceful Degradation of Transport Layer in Mobile Internet	120
<i>Y. Matsushita, T. Matsuda, M. Yamamoto</i>	

Session 4: Architecture, Implementation, and Experimentation

Implementing Ad Hoc to Terrestrial Network Gateways	132
<i>J. McGee, M. Karir, J.S. Baras</i>	
Connecting Wireless Sensornets with TCP/IP Networks	143
<i>A. Dunkels, J. Alonso, T. Voigt, H. Ritter, J. Schiller</i>	
Experimental Analysis of Heterogeneous Wireless Networks	153
<i>G. Iannello, A. Pescapè, G. Ventre, L. Vollerò</i>	

Session 5: Network and Protocol Modeling

High-Level Behavioral SDL Model for the IEEE 802.15.3 MAC Protocol .	165
<i>D. Dietterle, I. Babanskaja, K. Dombrowski, R. Kraemer</i>	
Performance of Phase Modulated Systems Using Fully Saturated Power Amplifiers	177
<i>Q. Lu, Q. Liu</i>	
On Connectivity in Ad Hoc Networks under Jamming Using Directional Antennas and Mobility	186
<i>G. Noubir</i>	
The Ad Hoc On-Demand Distance Vector Protocol: An Analytical Model of the Route Acquisition Process	201
<i>M. Hollick, J.B. Schmitt, C. Seipl, R. Steinmetz</i>	

Session 6: Wireless Network Scheduling and Analysis

A Scheduling Algorithm for a QoS Based Satellite Network	213
<i>A. Saha</i>	
A Power-Allocation-Combined Scheduling Algorithm for CDMA-Based High-Rate Packet Data Systems	225
<i>I. Koo, J. Zander, K. Kim</i>	
Analyzing the Performance of Data Users in Packet Switched Wireless Systems with Prioritized Voice Traffic	236
<i>R. Srinivasan, J.S. Baras</i>	

Session 7: Multimedia Distribution and Group Communication

Adaptive Multimedia Streaming for Heterogeneous Networks	248
<i>J. Korhonen</i>	

A Scalable and Adaptive Key Management Protocol for Group Communication	260
<i>Y. Challall, H. Bettahar, A. Bouabdallah</i>	
Smooth Fast Broadcasting (SFB) for Compressed Videos.....	272
<i>H.-F. Yu, H.-C. Yang, Y.-M. Chen, L.-M. Tseng, C.-Y. Kuo</i>	
Session 8: Service Discovery	
Dynamic Management of UDDI Registries in a Wireless Environment of Web Services	284
<i>Z. Maamar, H. Yahyaoui, Q.H. Mahmoud, F. Akhter</i>	
An Agent-Based Architecture for Service Discovery and Negotiation in Wireless Networks	295
<i>E. Bircher, T. Braun</i>	
Author Index	307

Distributed MAC Protocol to Improve Energy and Channel Efficiency in MANET

Chien-Yuan Liu¹ and Chun-Hung Lin

Department of Computer Science and Engineering,
National Sun Yat-Sen University, Kaohsiung 804, Taiwan
cyliau@csu.edu.tw, lin@cse.nsysu.edu.tw

Abstract. Bandwidth and battery power are key constraints for efficient and continuous operation of mobile computers. Power consumption during data transmitting and receiving significantly depends on the MAC protocol of wireless networks. This paper presents a channel efficient and power conserving protocol, named PC-DST, to enhance IEEE 802.11 DCF. PC-DST improves wireless channel throughput and saves huge amount of energy consumption. We illustrate PC-DST advantages via extensive simulation performed over wireless LAN. The results of simulation show that significant enhancements on frame goodput, frame delivery latency, and wireless channel efficiency are obtained. Moreover, PC-DST conserves more than 70% energy consumed in IEEE 802.11 DCF operation.

1 Introduction

The proliferations of portable computers and handheld devices have driven networks to support wireless connectivity [2, 12]. Wireless LAN (WLAN) is one of essential technologies of wireless computer networks. In fact, WLAN has successfully adopted in many campus networks and enterprise networks. Basically wireless networks deliver much less bandwidth than wired networks, for example 1-54 Mbps of WLANs versus 10-1000 Mbps of LANs. Thus, wireless medium is a scarce resource. When multimedia contents are broadly disseminated over WLAN, efficient utilization of wireless medium in WLAN is becoming an important issue [3, 5, 10].

To address the issue, a few past researches had proposed different protocols. Monks et al. [11] propose the power controlled media access (PCMA) protocol to improve channel utilization of wireless media. PCMA enables a greater number of simultaneous senders than the 802.11 by adapting the transmission ranges to be the minimum value required to satisfy successful reception at intended destination. Although, PCMA enhances the throughput of wireless media, a single hop wireless network is possible to become several multi-hop wireless networks due to transmission range reduced to its minimum. Multiple smaller ranges of wireless networks are formed by the space partition. On the other hand, power adjustment is

¹ The corresponding author is also an instructor in Electronic Engineering Department, Cheng-Shiu University, Kaohsiung 833, Taiwan

not only applied to medium access control of wireless ad hoc networks, but also affects the topology [14, 16, 17], the lifetime [2, 6] of wireless ad hoc networks.

Multi-hop wireless networks need a routing facility to find routes for delivering frames from source to destination. Perkins and Bhagwat [12] propose the destination-sequence distance-vector (DSDV) routing protocol for multi-hop mobile ad hoc networks (MANET). DSDV routing suffers from the propagation delay and the overhead of periodically updating its routing table. Next, Perkins and Royer [13, 15] propose the ad-hoc on-demand distance-vector (AODV) routing protocol which eliminates global periodic routing updates in DSDV. However, route discovery latency and per-hop processing overhead are essentially expenses.

We investigate the issues of efficient channel utilization and energy conservation at the MAC layer. Therefore, we propose a novel method, named PC-DST, for enhancing the channel utilization of 802.11 DCF. PC-DST adopts transmission range controlled medium access for spatial reuse purpose. In PC-DST, all communications follows a desired power constraint learned from previous handshaking, which guarantee that all transmissions would not disturb to each other during communication periods. This could enable multiple senders to simultaneously issue their communications within the same period. We model PC-DST mechanism and simulate communication behavior on randomly generated MANET. According to our simulations, the effect of enhancement is significant. PC-DST uses exact power requirement for transmitting frame from source to destination. Comparing to full power transmission of 802.11 DCF, PC-DST saves a large amount of energy consumed during data communications. To show the effectiveness of our scheme, we developed a simulation model to measure some transmission related data to calculate the channel efficiency and energy efficiency of PC-DST and 802.11 DCF. We compare the performance of energy utilization in the energy efficiency derived from the results of simulation. From simulation results, we assure PC-DST not only provides outstanding channel throughput, but also conserves a huge amount of energy for data communications in MANET.

The remainder sections of the paper are organized as follows. In Section 2, we introduce 802.11 DCF to be applied on MANET. Section 3 describes key ideas of PC-DST and the least-required power constraint for simultaneous transmissions. Section 4 describes our simulation model and compares the results of experimental simulation of PC-DST to that of 802.11 DCF. Finally, we summarize major results of the performance simulation in Section 5.

2 IEEE 802.11 Ad Hoc Networks

The IEEE 802.11 specification includes MAC layer and physical layer. This paper only addresses to the MAC layer portion. Aad et al. [1, 5, 7, 8] introduces 802.11 standard with more widespread descriptions. The detailed description is described in the ANSI/IEEE standard [18]. We briefly introduce the terms defined in IEEE 802.11 standard.

A Wireless Medium (WM) is the medium used to implement the transfer of protocol data unit (PDU) between peer physical layer (PHY) entities of WLAN. A

Station (STA) is any device that contains 802.11 conformant MAC and PHY interface to the WM. STAs working in either distributed coordination function (DCF) or point coordination function (PCF) form a basic service set (BSS). The BSS covered area is called the basic service area (BSA). A BSS can either be an infrastructure network or an independent ad hoc network.

An ad hoc network composed solely of stations within mutual communication range of each other via the WM. An ad hoc network is typically created in a spontaneous manner. The principal distinguishing characteristic of an ad hoc network is its limited temporal and spatial extent. These limitations allow the act of creating and dissolving the ad hoc network to be sufficiently straightforward and convenient so as to be achievable by non-technical users of the network facilities; i.e., no specialized technical skills are required and little or no investment of time or additional resources is required beyond the stations that are to participate in the ad hoc network.

There are two service control methods specified in 802.11 MAC. One is PCF and the other one is DCF. DCF provides contention based service, whereas PCF provides a contention free service. However, DCF is the only service provided in an ad hoc network. Therefore, we only focus on control function in this paper. DCF supports delay insensitive data transmission, and works in contention mode. 802.11 DCF adopts carrier sense multiple access (CSMA) / collision avoidance (CA) scheme. The hidden terminal problem [2] implies a collision is possible to happen while multiple hidden STAs try to transmit their frames after the channel to become idle. To avoid collision, 802.11 uses a binary exponential back-off scheme. The binary exponential back-off scheme is implemented by each station by means of a parameter, named back-off counter, which maintains the number of empty slots the tagging STA must observe on the channel before performing its own transmission attempt. When the tagging STA needs to schedule a new transmission, it selects a particular slot among those of the contention window, whose size is maintained in a MAC preset parameter CW_min . The back-off value is defined by the following expression:

$$Backoff_counter(rt_att) = \lfloor \{rand() \times \min(W_max, CW_min \times 2^{rt_att})\} \rfloor \quad (1)$$

Where CW_max is a MAC preset parameter, $rand()$ is a function which returns pseudo random number uniformly distributed in $[0..1]$, and rt_att is the number of retransmission attempt with initial value one. After each unsuccessful transmission, the rt_att is incremented by one. The STA doubles the contention window size until it reaches the CW_max . The increasing of the contention window size is the reaction that the IEEE 802.11 DCF provides to react to a congestion condition and to make the access adaptive to channel conditions.

The back-off counter is decreased as long as an idle slot is sensed, it is frozen when a transmission is detected, and reactivated after the channel is become idle for at least a DIFS time. While the back-off counter reaches the value zero, the STA can transmit its data frame. If the transmission generates a collision, the size of contention window is doubled for next retransmission attempt to reduce the contention of the medium.

RTS/CTS scheme is added to relieve the hidden terminal problem of the basic CSMA/CA scheme. RTS is sent before PDU transmission. If collision happens, the wasted channel time is only 20 octets instead of a full PDU length. CTS is replied by

a destination if it is ready to receive PDU. When source receives CTS, it starts transmitting its PDU. All other STAs update their network allocation vector (NAV) whenever they hear RTS, CTS, or PDU frame. The handshake-timing diagram of RTS/CTS/DATA/ACK, SIFS/DIFS, and NAV is shown in Fig. 1.

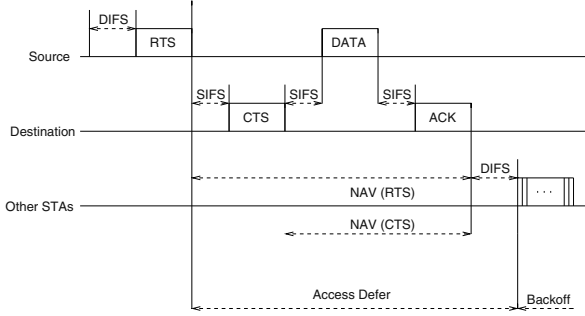


Fig. 1. DCF frame sequence

A source has to wait at least DCF Inter Frame Spacing (DIFS) time and an additional random back-off time after the channel is idle to resolve collision problem. Short ISF (SIFS) is shorter than DIFS. This is a simple prioritized scheme to let ACK, CTS, or PDU frame has higher priority than RTS frame.

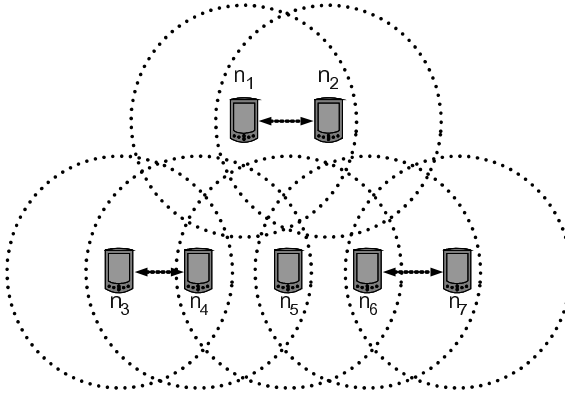


Fig. 2. Spatial reuse scenario

3 PC-DST Scheme

In a highly saturated ad hoc network, assume that all STAs in the network have power control functions in their radio units and all STAs are stably located within the coverage area which any STA can achieve each others with maximal transmitting power. Consider an example at a conference room with lots of mobile audiences,

when one audience starts a communication with full power, other users are depressed except for listening. In case that a source STA controls its transmitting power to reach the least-required level to its destination, the STAs outside the affected coverage area of the communication are possible to send their frames via the same channel. A snapshot of the simultaneous communications is shown in Fig. 2. The frame sequence of the scenario is depicted in Fig. 3, there are 3 extra frames sent over the links (n_3, n_4) and (n_6, n_7) during the period of primary communication (n_1, n_2) .

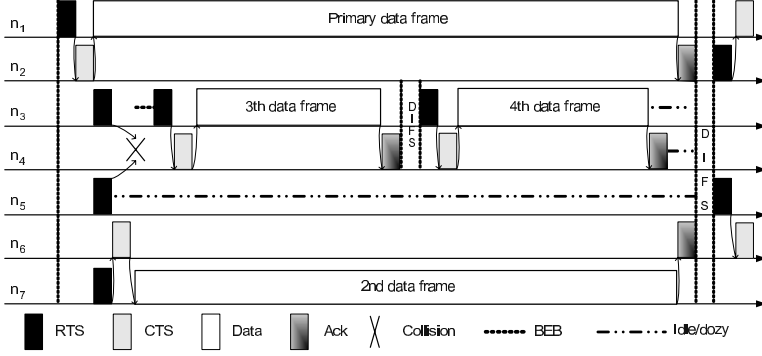


Fig. 3. Simultaneous frame sequence

The collision problem possibly arisen from the simultaneous communications can be resolved by adopting the binary exponential back-off (BEB) algorithm [2, 5], however, it degrades the channel utilization of the wireless communication channel [9, 11]. Therefore, we develop an efficient simultaneous communications scheme to solve the problem. The PC-DST method is depicted as follows,

- Like 802.11 DCF protocol, the source-destination pair initially uses RTS/CTS frames with maximal power to create a primary link. During this period, receiving side calculate the desired power $P_{desired}$ from equation (2) [9, 19] for further DATA/ACK frame transmissions. Let P_t be transmitting power level, P_r be received power level, Rx_{thresh} be necessary minimal receiving signal strength characterized by the receiver, and c be a tunable constant, then

$$P_{desired} = \frac{P_t}{P_r} \times Rx_{thresh} \times c \quad (2)$$

- $P_{desired}$ is formatted as an extended option and is piggyback to corresponding sender/receiver for closed-loop power control. The option field is depicted in Fig. 4, and is attached at the end of RTS/CTS/DATA/ACK frame. Code in the option denotes the type of extension function. In PC-DST, the code=1 specifies power control function. In this way, some STAs implemented with 802.11 would ignore the option tail, whereas the others implemented with the option extension would interpret the option tail to the transmitted P_t and $P_{desired}$. Thus, backward compatibility to 802.11 can be guaranteed.

- The neighbors in the coverage area of the primary link at P_t would go into dozy mode during the period of primary communication, the dozy interval is denoted by NAV. Under the dozy state, STA consumes only tenth of power in idle state.

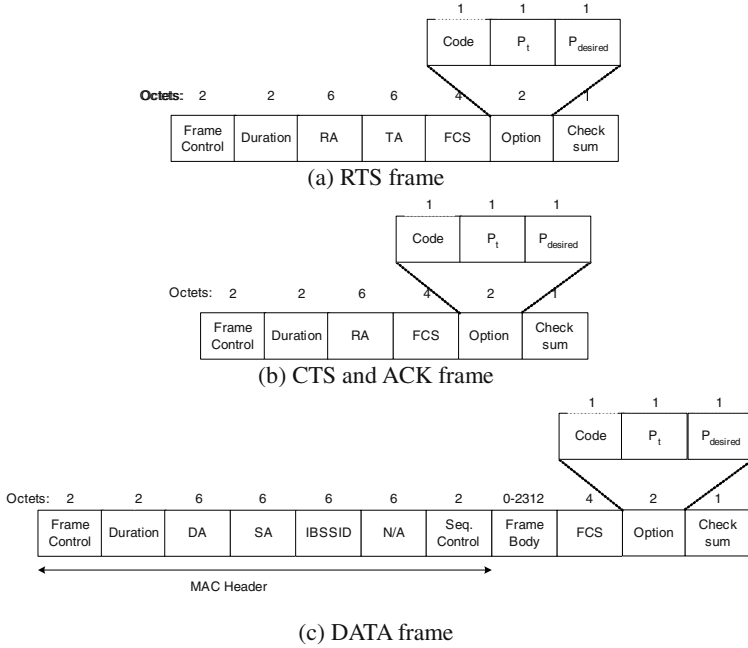


Fig. 4. Extended frame formats

- The STAs outside the coverage area are able to send frames, if
 - Data frames are waiting in their sending buffers
 - The newer links are connectible by RTC/CTS with restricted power level less than $P_{constraint}$ calculated from the overheard frames of the primary link or other simultaneous communications, and would not disturb them.
 - Binary exponential back-off algorithm is adopted to resolve possible collisions between the RTC/CTS of new generating links for simultaneous transmissions.
 - The interval of the newer links will not extend over the length of the primary-link duration. The reason is that, when dozy STAs wake up and try to send frames after the end of the primary link, they easily collide with former-newer links if one of them sends RTS at maximal power. This is because they can not sense that some prolonged links are working at restricted power level.

Through overhearing or normal communications, WA (weighted average) $P_{desired}$ together with corresponding MAC ID is locally kept in its local table for future heuristic prediction. $WA P_{desired}^{(t)} = \alpha P_{desired}^{(t)} + (1-\alpha) P_{desired}^{(t-1)}$, where α is a constant parameter and is set to 0.8 in our simulation. Note that the improvement of PC-DST scheme would be unclear as historical information is obsolete when mobile STAs rapidly move. In fact, most of on-demand algorithms for wireless ad hoc networks are unable to cope with rapid movement condition. However, PC-DST is still suitable for stable or slowly moving environments.

For instance, a sender firstly transmits RTS at power level of $P_t = P_{max} = 10$ and $P_{desired} = 10$, to inform overhearing STAs that my following transmissions will use this power and your possible transmission should not interfere to me with referring the power information carried in option tail. Next, the recipient replies CTS at power level of $P_{max} = 10$ and $P_{desired} = 6$, to inform the sender its receiving strength requirement and also to inform overhearing STAs. Then, the sender transmits data frame at power level of $P_t = 6$ ($< P_{max} = 10$) to shrink the coverage range and to conserve energy, and $P_{desired} = 5$ to inform its receiving strength requirement to the recipient. In the same way, the recipient replies ACK at power level of $P_t = 5$ and $P_{desired} = 6$. Thus, closed-loop power control can be achieved. This scenario also demonstrates asynchronous channel effect can be dealt with PC-DST, e.g. the sender and the recipient transmit with different power levels of 6 and 5 to reach their destinations under asynchronous channel condition, respectively. As for other STAs, they record the minimum power constraint, e.g. $P_{constraint} = 4$, from calculation of equation (2) with option information from overheard frames. If they want to access the channel, they firstly look for the destination entry in its local table to get WA $P_{desired}$ for trying to send data to the destination. If the WA $P_{desired} = 3$, for example, is less than the minimum power constraint $P_{constraint} = 4$, then the communication will be very possible to be established as a simultaneous communication. In case that the entry isn't existed yet, the communication would be postponed until the end of primary communication. After the end of primary communication, all transmission attempts again compete for access the channel with BEB contention resolution such as the beginning of this instance.

With this manner, multiple simultaneous communications are possible to be created without mutual interference. If sensing zone effect is considered together, PC-DST is still workable. When a STA senses certain signal from other communications and can't decode the message successfully, it will conservatively assume that it is located within others coverage and will keep in idle mode until the communication finished. Of course, power conservation is not applied in this case as the STA can't decode frame format to obtain the duration from the unknown communication in order to set its NAV correctively.

4 Simulation

To evaluate the performance of PC-DST scheme, we have developed a simulator using C. In simulations, the transmission radius is normalized in one by one square area, and each topology has 50 STAs generated randomly in the square region. Let the bandwidth of each STA be 1 Mbps and the power level of radio unit can be continuously controlled. Initially assume that the ad hoc network is static for proving the correctness of PC-DST method and consider mobility in future research. Various traffic loads are simulated by tuning mean frame inter-arrival time. Mean frame inter-arrival time shorter than 5 mini second (ms) represents heavy traffic load. Frames are generated with random sizes ranged from 64 bytes to 1024 bytes at random STAs. Related simulation parameters are listed in table 1.

The simulations are divided into two parts. One is the measurement of the bandwidth improvement of PC-DST to that of 802.11 DCF. The results of bandwidth simulation are described in Section 4.1. The other one is to measure the power conserving effect of PC-DST in comparing to that of 802.11 DCF. Section 4.2 illustrates power conservation in detail.

Table 1. Simulation parameter setting

Parameter name	Parameter value
Network dimension	1×1
# of STAs	50
Channel bandwidth	1 Mbps
Frame length	64 - 1024 bytes
Mean frame arrival time	2.88 - 5.76 mS
Topologies	200
Simulation time	300 S
Slot time	50 uS
RTS	20 bytes
CTS, ACK	14 bytes
DIFS	128 uS
SIFS	28 uS

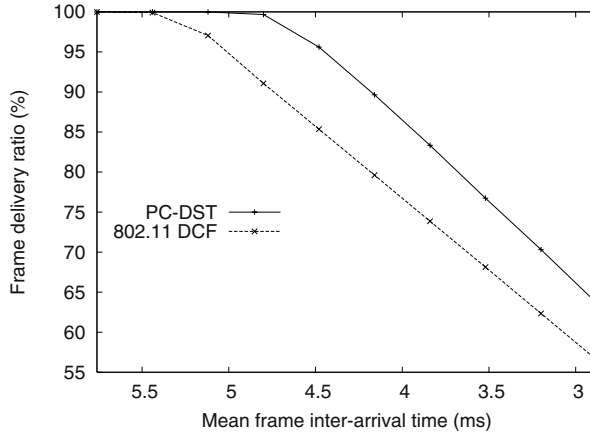


Fig. 5. Goodput vs. traffic load

4.1 Channel Efficiency Simulation

Fig. 5 shows frame delivery ratio (goodput) versus traffic load in mean frame inter-arrival time. The frame delivery ratio is denoted as ratio of the number of all successful transmitted data frames to the number of all generated data frames. In case of light traffic load, almost all generated frames can be delivered through wireless channel. Frame delivery ratio gradually decreased as traffic load increased. PC-DST sustains better ratio in heavier traffic load than 802.11 DCF. When mean frame inter-arrival time is at 4.5 mini seconds, frame delivery ratio of 802.11 DCF is near 86

percent, whereas PC-DST still maintains at more than 96 percent frame delivery ratio. The frame delivery ratio drops sharply while the traffic load is higher than that of 4.5 ms of the mean frame inter-arrival time. In heavy traffic load case, e.g. when mean frame inter-arrival time was at 3.5 mini seconds, frame delivery ratio of 802.11 DCF heavily dropped down to 68 percent, whereas PC-DST still kept its frame delivery ratio above 76 percent. The greatest improvement of frame delivery ratio by using DST was up to 12 percent in comparing to that of the standard IEEE 802.11 DCF.

Fig. 6 illustrates the mean frame delay time of 802.11 DCF and that of PC-DST. From Fig. 6, we know that PC-DST apparently shortened mean frame delay time in all cases, especially in heavy load situation. For example at mean frame inter-arrival time was at 3.5 ms, 802.11 DCF causes about 3.2 seconds of mean frame delay time, whereas PC-DST causes only 2.2 seconds. The improvement is up to 30 percent reduction of the mean frame delay time. In the growing applications of multimedia contents and real-time messaging over wireless communications, the enhancement of frame latency time has a great benefit for accelerating response time and for guaranteeing faster interaction of these kinds of applications.

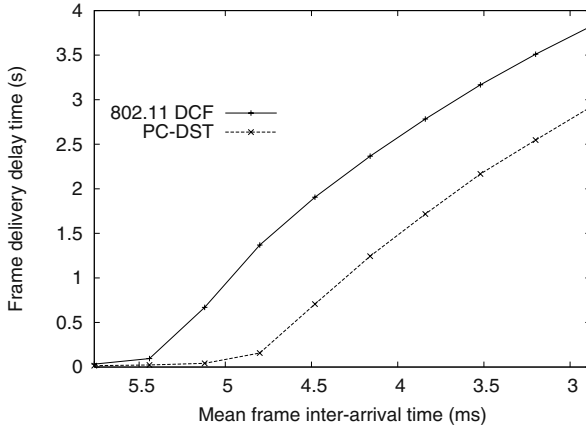


Fig. 6. Latency vs. traffic load

Channel efficiency is denoted as total transmitted data bits divided by total simulation time in micro second (μ s). Fig. 7 shows the bandwidth efficiency of PC-DST and that of 802.11 DCF. In case of light load, both bandwidth efficiencies are lower than 0.8 bits per second. This is because there is less data frames waiting for transmission. Therefore, channel is idle in most of time. The channel efficiency is increasing as the traffic load becomes heavier. However, 802.11 DCF reaches at saturation value of 0.82 bits per micro second. Thereafter, it decreases slightly as the traffic load keeps increasing. The reason of this situation is that more frame arrivals result more collisions, which waste channel bandwidth. In contrast, PC-DST mechanism still keeps its channel efficiency in increasing until it reaches at higher saturation value of greater than 0.92 bits per micro second. In our simulation, the maximum channel bandwidth is 1.0 bits per micro second. PC-DST mechanism performs very close to the upmost value of the channel bandwidth.

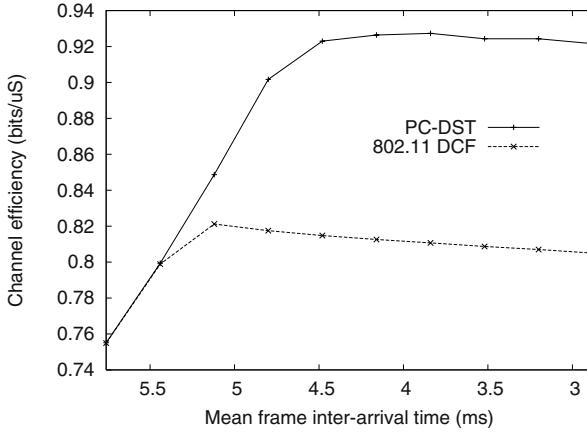


Fig. 7. Channel efficiency vs. traffic load

We summarize the bandwidth simulation as follows. PC-DST tries to send more frames by reusing space partition for more simultaneous transmissions. Therefore, PC-DST mechanism has higher frame goodput, lower frame delivery latency, and much higher channel efficiency.

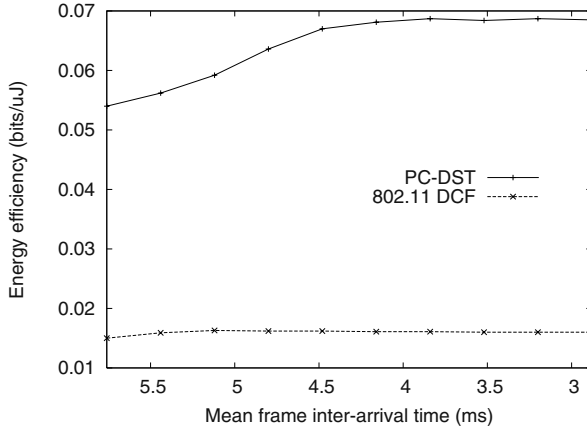


Fig. 8. Energy efficiency vs. traffic load

4.2 Energy Efficiency Simulation

Finally, we describe the energy efficiencies obtained from 802.11 and PC-DST. The energy efficiency is denoted as the total transmitted data bits divided by the total energy consumed during the whole simulation period. Fig. 8 shows that the energy efficiency of PC-DST is much higher than that of 802.11 DCF. This is due to that PC-

DST transmits more data frames than 802.11 DCF. In addition, PC-DST conserves more energy by transmitting frames with least-required power level and setting all covered stations into sleep mode. Hence, PC-DST conserves a larger amount of energy in comparison with that of 802.11 DCF. Therefore, PC-DST is more energy efficient. In light load cases, it provides nearly 4-times energy efficiency than 802.11 DCF. In heavy load cases, PC-DST further outperforms 802.11 DCF in energy efficiency. For example, at 3 ms of the mean frame inter-arrival time, the energy efficiency of PC-DST is near 5-times of that of 802.11 DCF.

5 Conclusion

In wireless ad hoc networks, channel bandwidth and battery energy are scarce resources. We proposed PC-DST scheme for enhancing frame goodput, frame delivery latency, channel efficiency, and energy efficiency of a highly saturated wireless ad hoc network. PC-DST enables multiple simultaneous communications by spatial reuse obtained from the control of radio power and proper end-time restriction. The option extended from standard frame format is proposed to realize the closed-loop power level control. The extension also considers backward interoperability to 802.11 DCF. To prove the method, we designed and developed a simulator to investigate the behavior of PC-DST scheme. Simulation experiments are performed in randomly generated topologies over various traffic loads. From the results of simulation, we confirm that PC-DST significantly improve goodput, latency, channel efficiency, and energy efficiency of wireless ad hoc networks. Moreover, PC-DST is a distributed scheme by using only locally historical and overheard information, and can be applied onto other prevalent MAC protocols such as HyperLAN or Bluetooth.

References

1. Aad and C. Castelluccia, "Differentiation mechanisms for IEEE 802.11," *Proceedings of IEEE INFOCOM 2001*, pp.209–218, vol.1.
2. V. Bharghavan, A. Demers, S. Shenker and Lixia Zhang, "MACAW: A Media Access Protocol for Wireless LAN's," *Proceedings of SIGCOMM 94*, pp.212–225.
3. L. Bononi, M. Conti and L. Donatiello, "Design and performance evaluation of a distributed contention control (DCC) mechanism for IEEE 802.11 wireless local area networks," *Proceedings of WOWMOM 1998*, pp.59–67.
4. L. Bononi, M. Conti and L. Donatiello, "A distributed mechanism for power saving in IEEE 802.11 wireless LANs," *ACM J. Mobile Networks and Applications*, vol. 6, pp.211–222, 2001.
5. F. Cali, M. Conti and E. Gregori, "IEEE 802.11 protocol: design and performance evaluation of an adaptive backoff mechanism," *IEEE journal on selected areas in communications*, vol. 18, no. 9, September 2000.
6. J.-H. Chang and L. Tassiulas, "Energy conserving routing in wireless ad-hoc networks," *Proceedings of IEEE INFOCOM 2000*.
7. B.P. Crow, I. Widjaja, J.G. Kim, and P.T. Sakai, "IEEE 802.11 wireless local area networks," *IEEE Communication Magazine*, September 1997.

8. Ganz, A. Phonphoem, and Z. Gaze, "Robust superpoll with chaining protocol for IEEE 802.11 wireless LANs in support of multimedia applications," *ACM Wireless Networks*, vol. 7, pp.65–73, 2001.
9. E.S. Jung and N.H. Vaidya, "A Power Control MAC Protocol for Ad Hoc Networks," *Proceedings of Mobicom 2002*, pp.36–47.
10. V. Kanodia, C. Li, A Sabharwal, B.Sadeghi, and E. Knightly, "Distributed multi-hop scheduling and medium access with delay and throughput constraints," *Proceedings of ACM SIGMOBILE 2001*, pp.200–209.
11. J.P. Monks and V. Bharghavan and W.-M.W. Hwu, "A Power Controlled Multiple Access Protocol for Wireless Packet Networks," *Proceedings of IEEE INFOCOM 2001*.
12. C.E. Perkins and P. Bhagwat, "Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers," *Proceedings of SIGCOMM 94*, pp.234–244.
13. C.E. Perkins and E.M. Royer, "Ad-hoc on-demand distance vector routing," *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, New Orleans, LA, February 1999.
14. R. Ramanathan and R. Rosales-Hain, "Topology control of multiple wireless networks using transmit power control adjustment," *Proceedings of IEEE INFOCOM 2000*.
15. E.M. Royer and C.E. Perkins, "Transmission Range Effects on AODV Multicast," *ACM J. Mobile Networks and Applications*, 2000.
16. S. Singh and C.S. Raghavendra, "PAMAS- power aware multi-access protocol with signaling for ad hoc networks," *ACM Computer Communication Review*, pp. 5–26, July 1998.
17. R. Wattenhofer, L. Li, P. Bahl, and Y.-M. Wang, "Distributed topology control for power efficient operation in multihop wireless ad hoc networks," *Proceedings of INFOCOM 2001*.
18. "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *ANSI/IEEE Standard 802.11, Part 11*, 1999 Edition
19. W.L. Stutzman and G.A. Thiele, "Antenna Theory and Design," *John Wiley & Sons*, 1998.

CGGC: Cached Greedy Geocast

Christian Maihöfer, Reinhold Eberhardt, and Elmar Schoch

DaimlerChrysler Research And Technology,
Communication Systems (RIC/TC)
P.O. Box 2360, 89013 Ulm, Germany
{christian.maihoefer,reinhold.eberhardt,
elmar.schoch}@daimlerchrysler.com

Abstract. With cached greedy geocast, we propose an enhanced forwarding scheme for geocast especially in highly mobile ad hoc networks. We will show, that a cache for presently unforwardable messages during what we call the line forwarding, can significantly improve the geocast delivery success ratio, especially in sparse peopled networks with high node velocities. Therefore, we introduce and examine two separate caching strategies and present simulation results in conjunction with geocast. Our simulations inquire the delivery success ratio but also the trade-off in delivery delay.

1 Introduction

For all routing protocols in mobile ad-hoc networks, the major challenge is to find a route from the sender to the destination without any preconfigured information and under constantly varying link circumstances. In contrast to topology-based routing, the approach of position-based routing relies on geographic position information to deal with this problem. This means that all routing decisions, for example to which next node the packet should be forwarded, are based on the geographic destination data included in the packet. We distinguish between two basic classes of geographic forwarding: hop-to-multhop and hop-to-hop. In hop-to-multhop protocols one packet is sent to more than one next node, which means that the packet has to be duplicated and possibly reaches the destination more than once. This routing scheme results in a redundant delivery to a single destination or in the delivery to multiple destinations. In the other class the packet is routed only on exactly one, distinct way, resulting in a unicast delivery.

This paper considers geocasting, which is the transmission of a message to some or all nodes within a geographical area, called destination region. Geocast in position aware networks allows promising new services and applications. For example, in the automotive domain we aim at realizing virtual warning signs. Virtual warning signs are displayed inside the car, e.g. on the dashboard and can warn the driver about an accident after a blind corner, about an icy road, a wrong-way driver, a speed limit, and so forth. Virtual warning signs can easily be realized with geocast.

When designing protocols for the automotive domain, we must especially take care to be able to cope with high node velocities, which is typically not

considered in related work about ad hoc networks. Our proposed cache scheme is an extension of regular geocast protocols in order to deal with this situation of high velocities, i.e. constant neighborhood changes and unstable routing paths.

The geocast protocol that we rely on and extend in this paper uses both routing schemes explained above. As long as a packet is outside the destination region of a geocast packet, a network efficient hop-to-hop delivery is applied. Inside the geocast destination region, hop-to-multihop routing allows the delivery to all nodes of the destination region. In our geocast protocol [1,2] we divide the overall delivery process in exactly these two substantial parts, which on the one hand is called line-forwarding as long as the initial packet hasn't reached its destination region and on the other hand area-forwarding inside of this region.

To be able to select a single next node during line-forwarding, we use a beaconing system that allows constant neighborhood awareness. If a packet arrives, the routing layer on each relaying node is able to decide, based on the neighbored nodes' and the packets' destination position information, to which particular neighbor a message should be forwarded.

The most obvious decision scheme in this context is greedy forwarding. When using it, a message is always forwarded to that neighbor, whose position is nearest to the packet's destination, which is primarily defined through a position vector. This procedure should guarantee that a message is routed to the destination position in a minimum number of hops. But greedy forwarding has also one considerable disadvantage. If a node tries to forward a message while no neighbor is present that is closer to the destination than the relaying node himself — we call it a local maximum or dead end — this node is not able to continue the greedy forwarding at this point in time. In more detail, a *local maximum* is reached at a node n if no neighboring node of n is closer to the destination region than n . In case a local maximum is reached, the routing protocol must provide a recovery strategy if the message shall not be lost.

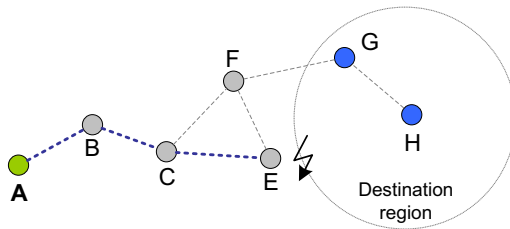


Fig. 1. Local maximum prevents successful forwarding

In Figure 1, node A sends a message to the depicted destination region using greedy forwarding. The described problem arises when the message reaches node E. Although there would be a logical path into the destination region through nodes A-B-C-F-G, simple greedy forwarding fails to deliver the message. For geocast it is especially reasonable to try to achieve a maximum of successful

line-forwardings, since each loss of a packet that doesn't reach the destination region prevents a successful delivery to any node inside of it.

In order to overcome these flaws, we propose cached greedy geocast. The main idea of cached greedy geocast is to add a small cache to the routing layer that holds those packets a node cannot forward instantly due to a local maximum. As we use a beaconing subsystem anyway to be able to discover and hold a table of neighbored nodes, this cache is notified whenever a new neighbor comes into reach or an already known neighbor changes its position. Then the cache can check all currently stored messages whether they can be forwarded to a newly discovered or moved neighbor, because those neighbor's position is now possibly closer to a geocast destination region than the current node's position.

Please note, our proposed caching scheme is especially designed for the use in ad hoc networks with high velocities and having certain applications in mind like the mentioned virtual warning sign. In particular, it is not meant to use that scheme for regular unicast delivery, since higher layer protocols like TCP are likely to result in degraded performance when caching is applied.

The overall paper is organized as follows: Section 2 discusses other approaches in the context of position-based routing, especially concerning geocast. In Section 3, we describe our proposed ideas and our implementation of cached greedy geocast. Afterwards, we will show simulation results of our algorithms and conclude the work with a brief summary.

2 Discussion of Related Work

For the classic definition of geocast, meaning that information is delivered to all nodes currently residing within a specific geographical region, quite a number of approaches have been proposed. Where the distribution of data inside the target area is mostly done in the same way by applying a flooding mechanism, the methods to transport data packets to that region differ elementarily. Basically, there are three methods implemented: restricted or modified flooding, hierarchical and greedy forwarding. For a complete overview of geocast protocols see [3].

Flooding derived protocols like Location Based Multicast (LBM) [4] or GeoGrid [5] usually define a virtual geographic region, inside of which every contained node forwards a packet, if it hasn't done so yet. Obviously, the protocol needs a sequence number mechanism to be able to identify packets that already have been received and forwarded at a node before. Because the forwarding region also includes the final destination region, it is assured that the data packets reach all concerned nodes within this destination region. A node residing inside the destination region then can pass the data to higher layers. The advantage of that delivery process is redundancy, since messages are able to reach the destination on many ways in parallel. This, of course, results in high network load typical for flooding mechanisms.

A completely different approach is to apply a hierarchical forwarding. In [6], an approach is proposed which uses geographic information for routing in a fixed network like the internet. The GeoNode protocol assumes that the network has a cellular structure. For each cell, there is a node called GeoNode that relays

all messages for the nodes inside the cell. Position-aware GeoRouters take care to distribute data to other GeoRouters according to the packets' geographic destination information. If a node wants to send a message, it forwards it to the local GeoNode, which in turn passes it to the next GeoRouter. The GeoRouter then determines to which other GeoRouters the packet has to be forwarded in order to cover the complete destination region.

The other approach in this context is GeoTORA [7]. The basic idea of this approach is the usage of an acyclic, directed graph that allows to route the packet by corresponding graph algorithms. GeoTORA maintains such a graph for every intended geocast destination region. The graph representation is implemented by assigning a height value to every node on the network, where nodes inside the destination region carry a value of zero. All other nodes' height values are assigned according to the number of hops that are necessary to reach the destination region. When forwarding a message, a node always transmits it to a neighbor with smaller height.

As we already described, we used the third possibility, greedy forwarding, for our own implementation. Reasons for this decision are that it neither has the effect of the high network load caused by a flooding approach nor the necessity of building up a hierarchical structure. This makes it a good candidate for our considered highly mobile ad hoc networks.

Nevertheless, pure greedy forwarding also has the flaw that it cannot reach a destination if a local maximum at a forwarding node is reached. This issue is targeted for example in GPSR [8]. GPSR is a position-based unicast routing protocol which generally applies greedy forwarding and introduces a so called "perimeter mode" to route around a gap in the network. As long as a packet cannot be forwarded to a node closer to the destination, it surrounds the network gap on a tangential way. The idea of perimeter mode is to forward the message according to the right-hand-rule. This means, node x selects those neighbor n as next hop, which is the first in counter-clockwise order to a thought line from node x to the destination position. This method is only used if no closer neighbor is available.

Our approach described in this paper also targets this greedy forwarding flaw by caching packets that cannot be forwarded instantly. This solution is especially effective in highly mobile and sparse peopled ad hoc networks, where the perimeter mode of GPSR is less effective (see simulation results in Section 4). As examined in [9], node mobility can increase the overall capacity of an ad hoc network. We will show that cached greedy forwarding can also utilize the mobility to significantly improve delivery success probability.

3 Overview of Cached Greedy Geocast

3.1 Detailed Problem Description

Geocast in its most general definition means to transport a packet from the sender to all nodes that currently are inside a given geographic region. We assume that the sender specifies the destination region he wants to send the packet to. Therefore, we only need position and neighborhood information for routing. The

region definition is variable, but contains at least one point inside of the region, which we use as target for our routing. Usually, a circular area with its center and its radius is applied. As introduced before, we decided to use a greedy mechanism that routes a packet hop per hop in direction to the destination region, until the packet is received by a node that is inside this region. Then, an area-forwarding method, flooding, is applied to spread the packet among all nodes inside the region. The main focus of the paper is the line-forwarding part of geocast.

At first, simple greedy forwarding has some major disadvantages. Because the routing scheme relies on the availability of suitable neighbors, i.e. at least one neighbor that is closer to the destination, it is necessary to define what takes place if this elementary condition is not given. Usually, a packet is dropped under these circumstances. But this results in many losses, since network gaps, partitions, and outdated neighbor information are quite common in highly mobile scenarios. Moreover, although our geocast is a "best effort" service, we cannot accept such high loss probabilities, because any packet that is dropped during the line forwarding cannot reach *any* of its target nodes inside the destination region. On the other hand, using perimeter mode of GPSR, our simulation results (see Section 4) show that it is not effective with high velocities.

We define in the following two strategies which improve line-forwarding with the simple greedy scheme: enhanced and cached greedy forwarding. Both techniques profit explicitly from mobility, which we commonly face in our applications. The main focus of this paper is the caching approach, but we also describe and evaluate the enhanced forwarding shortly to be able to estimate its effects.

3.2 Enhanced Greedy Forwarding

A simple recovery strategy for greedy forwarding failures is to delay a packet a short period of time, and then retry to forward it. If this neighbor is no longer within the wireless transmission range, another neighbor is selected. This procedure may be repeated some times. If a suitable neighbor is in reach at any retry time, the packet is finally forwarded, or otherwise dropped, too. Although this is a quite simple scheme, it already helps in highly mobile ad hoc networks to improve delivery success ratio.

3.3 Cached Geocast Approach

In order to significantly achieve further reduction in experienced packet losses during line-forwarding, we propose a caching mechanism. The idea of cached geocast is to add a storage to the routing layer that takes up those packets, where the basic greedy schemes have failed. This cache of currently unroutable packets, called *LocalMaxCache* doesn't make any forward attempts himself, but gets notified about each newly discovered neighbor or about changes in neighbors' positions. Neighbors are discovered and maintained by a periodic beaconing system (see Figure 2), and the information in every beacon also contains the current position of the node the beacon comes from. Thereby, whenever having detected a new neighbor or changed relative positions to neighbors, the beaconing subsystem can notify the LocalMaxCache and provide the relevant position

```

N gets beacon (S, P):
// S...address of beacon sender, P...position of sender
// T...neighbor table
(1) if  $\exists i : T[i].address == S$  then // neighbor already registered
(2)    $T[i].position := P$ ; // update position
(3) else // neighbor is new
(4)   add  $(S, P)$  to  $T$ ;
(5) checkCache; // check cache for forwardable messages

```

Fig. 2. Pseudocode for beaconing system

information. The cache checks, whether there is a packet stored, whose destination fits to the new node in that manner that the new node is closer to it (see Figure 3). This means, our cache operates on demand and does not result in higher network message overhead in contrast to blind periodical resending attempts. The complete forward algorithm in pseudocode is given in Figure 4.

We expect from the caching strategy to significantly increase geocast delivery ratio especially in highly mobile environments. A trade-off will be to experience higher end-to-end delays. However, note that only packets are delayed which are currently not forwardable and which would be dropped otherwise. This means, the cache does not increase delay for those packets that would be delivered with regular greedy routing, too.

3.4 Implementation Description

We propose and investigate two separate alternatives for the line-forwarding cache. The first one consists of a size-restricted queue for packets, where *size* limits the number of packets stored and not the actual payload size. The length of the storage period is unlimited. The other approach hasn't a limitation of size, but packets will only be stored for a limited period of time.

Size restricted cache

After a packet has passed all attempts to forward it to a node that is closer to its destination region than the current relaying node, it is handed over to the cache. If the cache queue is already full, i.e. the maximum size is reached, the oldest packet in it is dropped which means that its second chance to reach

```

N procedure checkCache:
// Q...LocalMaxCache queue
// g...geocast message, c...geocast destination region center
// T...neighbor table
(1) if  $\exists i, j : \|pos(T[j]), Q[i].c\| \leq \|pos(N), Q[i].c\|$  and
(2)    $\forall k : \|pos(T[j]), Q[i].c\| \leq \|pos(T[k]), Q[i].c\|$  then
(3)   send geocast  $Q[i].g$  to  $T[j]$ ;
(4)   remove  $Q[i]$  from  $Q$ ;

```

Fig. 3. Pseudocode for checking the cache about forwardable packets

```

N forward packet (P):
// P.d...packet's destination region, P.d.c... destination region center
// P.fwdRetries...packet's forward attempts at current node
// maxFwdRetries...maximum forward retries
// T...neighbor table, nh...next hop
(1) nh := findNearestNeighbor(P.d.c); // find neighbor closest to destination
(2) if nh != NULL then // suitable neighbor was found
(3)   send packet P to nh
(4) else // no greedy neighbor available
(5)   if P.fwdRetries < maxFwdRetries then
(6)     delay packet P and recall forward(P);
(7)   else
(8)     put packet P into cache;

```

Fig. 4. Main forwarding mechanism

the destination region ends at that time. It is obvious, that dropping packets at the end of the queue implicitly limits the caching time depending on how much traffic occurs. On the other hand, in low traffic situations, packets may remain in the cache very long or even until the node is powered off. As long as a packet is stored in the cache, it is checked against every discovered neighbor, whether this one is a suitable next relay for it.

Time restricted cache

The time restricted cache works similarly to the size restricted cache but takes queuing time into account. A new packet is always added to the queue, no other packet has to be dropped. All added packets are stored for a certain period of time and dropped after the expiration of this interval. Thereby, we can specify a border beyond of which it doesn't make sense to forward a packet any more. Another aspect of this restriction alternative is the possibly unlimited growth of the cache. However, if we define a maximum caching time and given a known maximum sending and receiving rate, the maximum cache size is deterministic and known. Besides of that, packet forwarding out of the cache is done in the same manner as described before.

4 Simulation Results

In order to evaluate our approach, we implemented our cached geocast in ns-2, and examined several preliminary expectations we had on the design as well as the behavior of cached greedy geocast in contrast to GPSR, which implements a different mechanism to overcome local maximums.

4.1 Environment and Setup Parameters

For our simulations, we considered several parameters as important. The most significant of them are of course network size and node count, which define the effective node density. Commonly, quadratic network sizes between 1000 and 4000 meters side length, with a 500 m interval, and a node count of 100 are

used. As a standard wireless transmission range 250 m were applied. Therefore, node density varies from an average of about 20 nodes in the wireless transmission range in a 1000x1000m network to statistically 1.2 nodes in the wireless transmission range when using a 4000x4000m network.

All nodes are initially distributed uniformly over the network and move according to the random waypoint model. As we deal with highly mobile networks, a maximum speed of 50 meters per second and a pause time of 0 are common to all simulations. Note that these network and mobility specifications represent quite hard conditions for an ad hoc routing protocol. But these conditions were chosen intentionally, since it is a goal of our approach to overcome and even profit from them. To prove this, we will also show a comparison of our approach and GPSR both with no node mobility, i.e. a static network.

Another relevant item is the traffic that is to be transmitted on the network. In every scenario, we send 100 messages. Both the sender and the destination region are selected randomly, whereby the destination region radius ranges from a minimum of 100 to a maximum of 300 meters. For the comparison simulations with GPSR, which needs a node address as destination, we randomly choose one node out of the destination region. Our implementation of geocast does not need a node address but only the destination region. As we use an average value of 20 single simulation runs with identical parameters except the traffic, all other effects to the results should be eliminated.

transmission range	250m
mobility model	random waypoint
maximum speed	50m/s
simulation time	60s
network size	1000m to 4000m squares
node count	100

Fig. 5. Overview on basic simulation parameters

4.2 Evaluation Results

The most interesting values to examine are the mainly targeted delivery ratio and the end-to-end delay.

Line-Forwarding examination

We start with simulations of pure line-forwarding without geocast delivery, i.e. unicast delivery, with different count of allowed forwarding retries if a suitable neighbor is not available. This addresses the enhanced forwarding and primarily doesn't touch the proposed caching. In this context two parameters are important: first one is the allowed retry count, second one is the interval between these forwarding attempts. To be able to estimate the resulting differences clearly, we chose one strict pattern similar to simple greedy forwarding that allows no retries. Besides that, we applied a scheme which allows 1 retry after 0.1 seconds

of waiting, and one more lax pattern allowing 5 retries and 0.5 seconds between them.

As depicted in Figure 6, a resulting gain in delivery success ratio through more retries is shown.

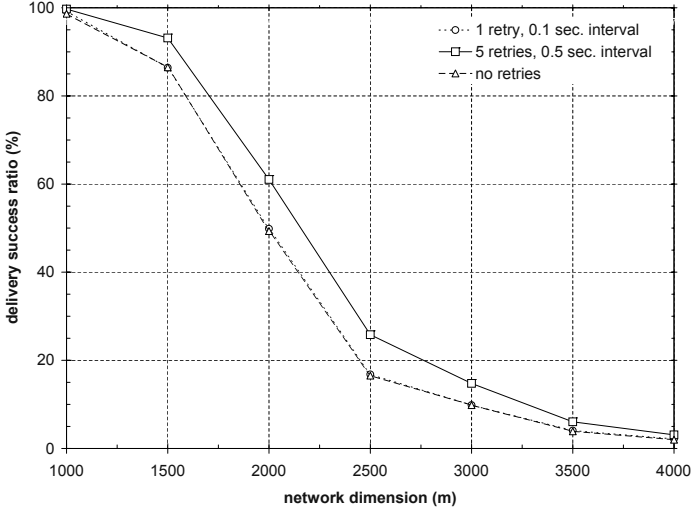


Fig. 6. Line-forwarding success rate results with different retry count and interval parameters

Having observed the effects of the enhanced greedy forwarding, we now focus on the cached forwarding mechanism. Therefore, we compare the line-forwarding as seen before with the cache scheme. A simulation result of GPSR, which has the same task as our line-forwarding, is given as a reference for comparison with our schemes.

In Figure 7, we see our expectations confirmed. The caching results in a significantly increased delivery success ratio in comparison to the ordinary line-forwarding. GPSR is already outperformed by our simple enhanced one retry line-forwarding in the more dense networks up to 2500 meters. The added caching outstrips both other schemes. Especially at network sizes over 2500 x 2500 meters, which are important for our applications as we assume that penetration rates may be low when introducing vehicular ad hoc networks, the caching approach performs very well.

For further investigation of the advantages and disadvantages of perimeter mode in contrast to our caching approach, it is necessary to examine the results of the opponent algorithms in mobile and non-mobile network scenarios. In Figure 8, it becomes clear that cached line-forwarding performs worse than GPSR with perimeter mode in non-mobile networks, since the cache works properly only with mobility. On the other hand, with the high velocities we assume for vehicular scenarios, cached line-forwarding reaches much higher delivery ratios than GPSR.

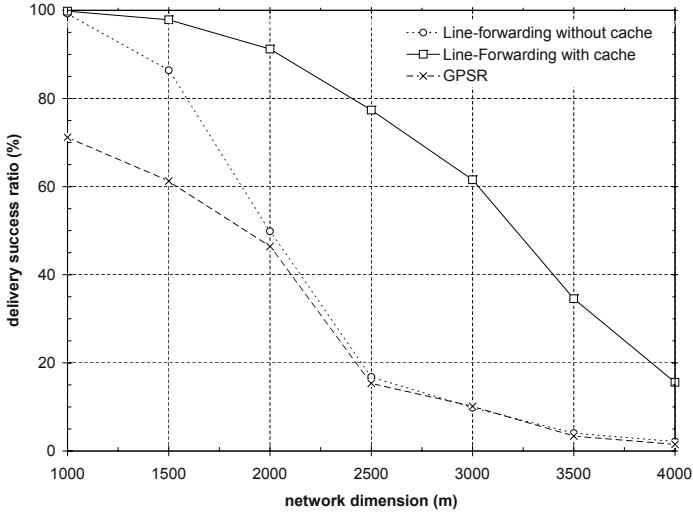


Fig. 7. Delivery success ratio comparison between standard line-forwarding, cached line-forwarding, and GPSR

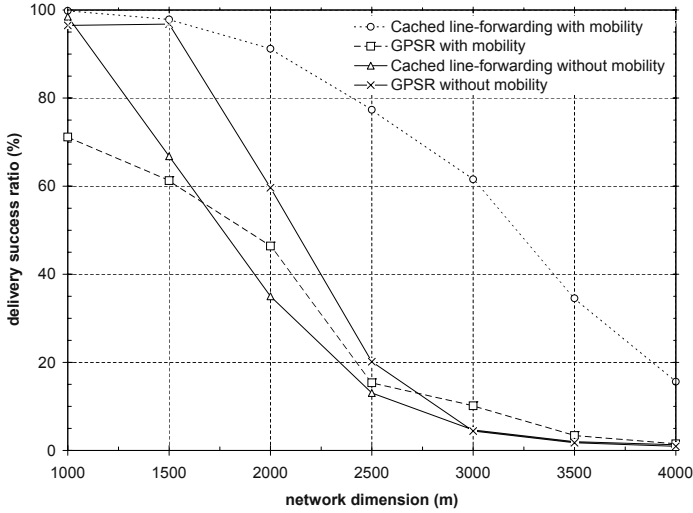


Fig. 8. Delivery success ratio of GPSR and cached line-forwarding in highly mobile as well as immobile ad hoc networks

Geocast with and without cache

Until now we have only seen results concerning the line-forwarding, which is of course the phase of geocast, where the cache can unfold its function. As we propose the approach in the context of geocast, we will now take a closer look on the impacts of caching on geocast.

We compare a standard geocast using only the enhanced forwarding scheme with our two caching approaches, size-restricted and time-restricted cache. In Figure 9, we display the simulation results.

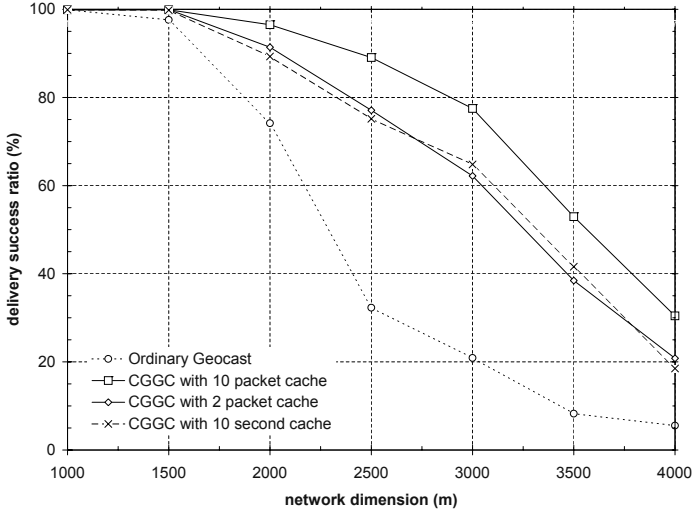


Fig. 9. The number of successful geocasts is increased by caching during the line-forwarding. The cache with 10 packets capacity performs best.

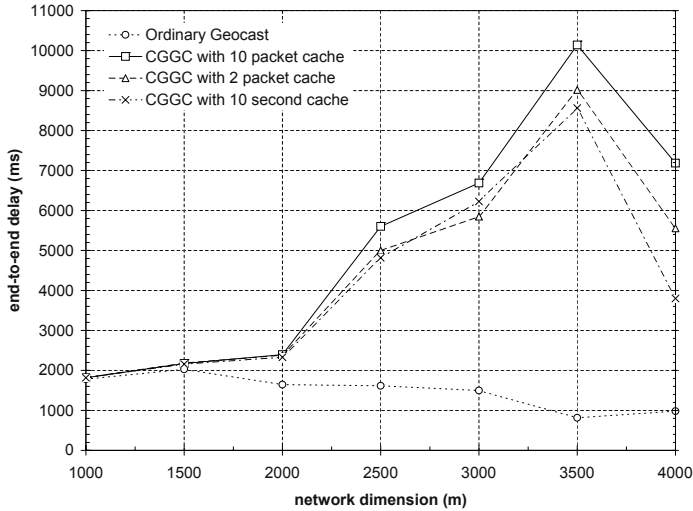


Fig. 10. Delays of cached geocast alternatives vs. ordinary geocast show the effect of caching.

These results allow several conclusions: Geocast profits from the gain in successful line-forwardings in the same magnitudes as line-forwarding alone. We see that the time-restricted cache performs slightly worse in this scenario than the size-restricted one. In these simulations, we used 10 packets as maximum size-restriction and a caching time of maximum 10 seconds on the other variant, respectively. Another very interesting aspect is the significant improvement of a small cache with only 2 packets compared with standard geocast. This means, even such a small cache is quite effective.

Concerning the protocol overhead, which we measured by the sent messages, there are only minimal effects of the caching scheme, since the cache is a on-demand mechanism and simply utilizes the beaconing that is running anyway.

Figure 10 shows another expected result of the cached greedy geocast approach: end-to-end delays are significantly increased. There is clearly a trade-off between delivery ratio and delay results — the implementation with the highest delivery ratio also shows the highest delay values, which of course is caused by the time a packet rests inside the cache. But it is important to realize that these high delays are average values caused by those packets that reach their destination region finally only due to the caching. From this point of view, the caching doesn't result in higher delays for those packets that do not need it, but increases the overall delivery ratio. Please note that smaller delay results for the 4000 m square network are caused by the lack of a representative number of successful deliveries.

5 Summary and Outlook

We have presented cached greedy geocast, a mechanism to improve greedy forwarding that is especially suitable for ad hoc networks with high mobility and high node velocities and low node densities, i.e. sparse networks. The work shows that both alternatives, size restricted and time restricted caching, can significantly increase the messages successfully delivered to a geographic destination region.

As a trade-off, we experience higher average end-to-end delays with caching. This is because more messages reach their destination region. However, these additional delays are only caused by those packets that reach their destination region only due to the caching mechanism and which would be dropped, otherwise.

As network load is concerned, the caching nearly doesn't have any effects, since it is a on-demand mechanism. The load is increased only slightly, because more line-forwardings can be continued successfully.

References

1. C. Maihöfer, Walter Franz, and Reinhold Eberhardt, "Stored geocast," in *Proceedings of Kommunikation in Verteilten Systemen (KiVS)*, Leipzig, Germany, Feb. 2003, pp. 257–268, Springer Verlag.

2. C. Maihöfer and Reinhold Eberhardt, "Time-stable geocast for ad hoc networks and its application with virtual warning signs," *Special issue of Computer Communications*, to appear 2004.
3. C. Maihöfer, "A survey on geocast routing protocols," *accepted for publication in IEEE Communications Surveys and Tutorials*, 2004.
4. Young-Bae Ko and Nitin H. Vaidya, "Geocasting in mobile ad hoc networks: Location-based multicast algorithms," in *Proceedings of the 2nd Workshop on Mobile Computing Systems and Applications (WMCSA 99)*, New Orleans, USA, Feb. 1999, pp. 101–110.
5. W.-H. Liao, Y.-C. Tseng, K.-L. Lo, and J.-P. Sheu, "GeoGRID: A geocasting protocol for mobile ad hoc networks based on GRID," *Journal of Internet Technology*, vol. 1, no. 2, pp. 23–32, Dec. 2000.
6. T. Imielinski and J. Navas, "GPS-based addressing and routing," Internet Engineering Task Force, Network Working Group, Request for Comments, RFC 2009, Nov. 1996.
7. Y.-B. Ko and N. H. Vaidya, "GeoTORA: A protocol for geocasting in mobile ad hoc networks," in *Proceedings of the 8th International Conference on Network Protocols (ICNP)*, Osaka, Japan, Nov. 2000, pp. 240–250.
8. B. Karp and H. T. Kung, "Greedy perimeter stateless routing for wireless networks," in *Proceedings of the Sixth ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom 2000)*, Boston, USA, Aug. 2000, pp. 243–254.
9. M. Grossglauser and D. Tse, "Mobility increases the capacity of ad-hoc wireless networks," *IEEE/ACM Transactions on Networking*, vol. 10, no. 4, Aug. 2002.

Design of Energy Efficient Wireless Networks Using Dynamic Data Type Refinement Methodology

Stylianos Mamagkakakis¹, Alexandros Mpartzas¹, Georgios Pouiklis¹,
David Atienza³, Francky Catthoor^{2,*}, Dimitrios Soudris¹, Jose Manuel Mendias³, and
Antonios Thanailakis¹

¹VLSI Design and Testing Center-Democritus University, Thrace, 67100 Xanthi, Greece
{smamagka, ampartza, gpouikli, dsoudris, thanail}@ee.duth.gr

²IMEC vzw, Kapeldreef 75, 3001 Heverlee, Belgium
Francky.Catthoor@imec.be

³DACYA UCM, Avda Computence s/n, 28040, Madrid, Spain
{datienza, mendias}@dacya.ucm.es

Abstract. This paper presents a new perspective to the design of wireless networks using the proposed dynamic data type refinement methodology. In the forthcoming years, new portable devices will execute wireless network applications with extensive computational demands (2 – 30 GOPS) with low energy consumption demands (0.3 – 2 Watts). Nowadays, in such dynamic applications the dynamic memory subsystem is one of the main sources of energy consumption and it can heavily affect the performance of the whole system, if it is not properly managed. The main objective is to arrive at energy efficient realizations of the dominant dynamic data types of this dynamic memory subsystem. The simulation results in real case studies show that our methodology reduces energy consumption 50% on average.

1 Introduction

Wireless communications have experienced a rapid growth over the latest years. The complexity of modern wireless networks is increasing, supporting a wide variety of services. Such complex systems require a combination of hardware and embedded software components in order to deliver the required functionalities at the desired performance level. Additionally, portable computers like PDAs and laptops using this wireless communication to interact with the environment rely on their limited battery energy for their operation. Energy consumption is the limiting factor in the amount of functionality that can be placed in these devices. More extensive and continuous use of network services by multimedia applications will only aggravate this problem.

Network applications are characterized by their various input and output streams, having different quality of service requirements. Depending on the service class and QoS of a connection, the memory footprint and accesses of these applications vary greatly. Therefore, the use of dynamic memory is imperative and must be in accordance to the dynamic behavior of the application. This behavior is often characterized

* Also professor at the Katholieke Univ. Leuven, Belgium

by complex algorithms that operate on large dynamically allocated stored data structures (i.e. single and double linked lists, arrays, dynamic first-in-first-out buffers) and are usually implemented with the use of the C or C++ programming language. The data are used for communication between multiple processes and may be shared among concurrent tasks. Adaptations to the dynamic nature of wireless networks are necessary to achieve energy efficiency and acceptable QoS.

The wireless network applications considered in this paper are encountered in the middle layer protocol processing and can be found from the MAC layer up to the transport layer of the OSI protocol stack. These are algorithms operating on large and irregular data structures, which are allocated and stored dynamically as dynamic queues, lists and association tables of records indexed by multiple keys, with the support of services to insert, locate, remove or substitute a record. Finally, stringent real-time requirements apply due to the very high bit-rate I/O data streams [1].

In this paper, we present a new methodology that allows developers to design wireless network applications with reduced memory energy consumption by utilizing the different dynamic data types (DDTs from now on) available in the construction of the dynamic memory subsystem. First, we define the relevant DDT search space, including the various DDTs, their combinations and their multilayered implementation. Then, we propose a way to find the dominant DDTs of the wireless network applications, explore all the available options in the search space and conclude to refined DDTs that will consume the least energy possible.

The remainder of the paper is organized as follows. In Section 2, we describe some related work. In Section 3, we define the search space of DDTs and some wireless network application behaviors. In Section 4, we present the DDT refinement methodology. In Section 5, we introduce our case studies and present the simulation results obtained. Finally, in Section 6 we draw our conclusions.

2 Related Work

The work presented in this article is inspired by [2], [3] and [4]. There are however three main differences. First of all, as opposed to optimizing heavy data-oriented multimedia applications (e.g. 3D Games, 3D rendering algorithms etc. [4]), this article focuses on wireless network applications. We show that by applying similar DDT refinements, wireless network applications consume significantly less energy.

The second difference is the software implementation of energy efficient DDTs as opposed to explicitly designing and using specific, configured, physical memories (hardware) [3]. We assume that the hardware is already designed and fully functional and that the refinement of the software will lower the memory energy consumption.

The third difference with the referenced work is that instead of focusing on the table lookups and on the accesses of the table keys [2], we explore all the DDTs of the network applications and the available search space in a more detailed way.

Optimizations and techniques for general purpose design to reduce energy consumption are explained in [5]. However, refining the DDTs at the software level in wireless network applications with complex dynamic behavior has not been given

much attention. Related work, on wireless network protocols [16] is supplementary to our work and supports our results.

In this paper, we propose using a fast, stepwise, cost-driven exploration and refinement for the DDTs in wireless network applications at the highest abstraction level, where the impact on memory performance and consumption is the most crucial.

3 Dynamic Data Type Search Space and Application Behaviors

3.1 Dynamic Data Type Search Space

In this subsection we introduce the DDTs available for our exploration and final refinement. These DDTs consist of various sets of data (records) put together, usually in the form of doubly linked lists [6]. They differ in the way these data types are interconnected and in the way that records can be added or subtracted during runtime to adjust themselves to the dynamic nature of the application. The DDT search space consists of the basic DDTs, their combinations and variations (as shown in Table 1).

Table 1. Abbreviations used throughout the text

Abbreviation	Explanation
CLS1WEL	“Embedded” single linked list of arrays
CLS1WPLO	“Pointer” single linked list of arrays with roving pointer
CLS1WPL	“Pointer” single linked list of arrays
CLS2WEL	“Embedded” double linked list of arrays
CLS2WPLO	“Pointer” double linked list of arrays with roving pointer
CLS2WPL	“Pointer” double linked list of arrays
CLSAR	Simple array
CLSPA	A pointer array of arrays
LL1WEL	“Embedded” single linked list
LL1WPLO	“Pointer” single linked list with roving pointer
LL1WPL	“Pointer” single linked list
LL2WEL	“Embedded” double linked list
LL2WPLO	“Pointer” double linked list with roving pointer
LL2WPL	“Pointer” double linked list

The basic dynamic data types are:

- **Array (CLSAR).** An array is a set of sequentially indexed elements having the same intrinsic data type, which is called element of the array. Each element of the array usually is a record of the application. All arrays consist of contiguous memory locations. The average access count to a random element is 1.

- **Single linked list (LL1WEL).** A single linked list is a set of data types or data structures that are connected with each other via pointers. Each element of the single linked list holds a record of the application and points to the memory address of the next element. The average access count to a random element is $(N/2 + 1)$, where N is the number of total elements.
- **Double linked list (LL2WEL).** A double linked list is a set of data types or data structures that are connected with each other via pointers. Each element of the double linked list holds a record of the application and points to the memory address of the previous and the next element. The average access count to a random element is $(N/4 + 1)$, where N is the number of total elements.

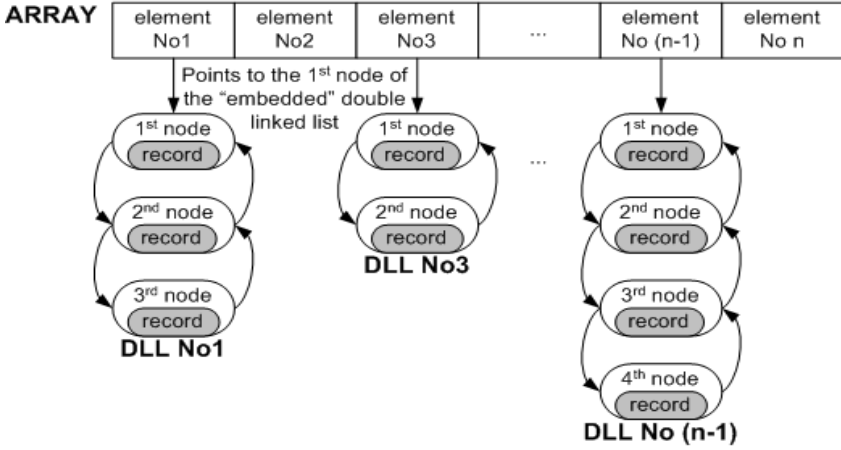


Fig. 1. An array of double linked lists with embedded records

These basic dynamic data types can have variations:

- **Embedded (EL).** In the embedded variation the record of the application is stored within the DDT. This means that the DDTs must be stored in bigger memories (to include memory space for the records) but no additional memory accesses are needed to read the records. An example of “embedded” records is shown in Fig. 1.
- **Pointer (PL).** In the pointer variation the record of the application is stored outside the DDT and is accessed with a pointer. This means that the DDTs can be stored in smaller memories but the records will need one more extra memory access (for the pointer) to read them. An example of “pointer” records is shown in Fig. 2.
- **Roving pointer (O).** The roving pointer is an auxiliary pointer to access a particular element of a list with less memory accesses [7]. If an application accesses its records sequentially, we can store the address of the element, which was accessed last, in a pointer and use it as a start for the next element access. This way, the access count to an element can be reduced drastically from half the list size down to two memory accesses.

Finally, the basic data types can be combined. In this way, we can have arrays of “embedded” doubly linked lists (as shown in Fig. 1), single linked lists of “pointer”

single linked lists (as shown in Fig.2), doubly linked lists of “embedded” arrays, arrays of “pointer” single linked lists with roving pointer etc. The difference between these complex dynamic data structures lie within the memory footprint and the memory accesses needed to access a random element.

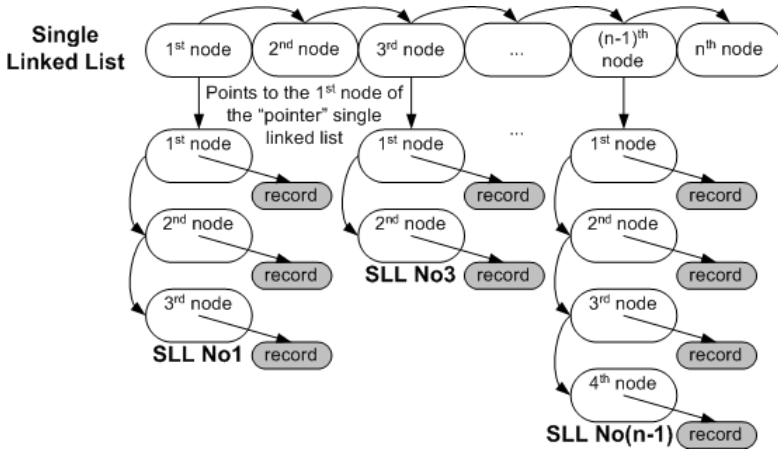


Fig. 2. A single linked list of single linked lists with “pointer” records

For example if we have 100 records, we can split them by using 2 keys instead of 1 and have 20 sets of 5 records. This means that we are transforming a single linked list (with 100 elements) to a single linked list (with 20 elements) of single linked lists (with 5 elements). So instead of having an average access count of 50 for a random element, we have now an access count of 12.5. The tradeoff comes of course with memory footprint (instead of allocating memory space for a simple 100-element data structure, we allocate memory space for a complex 120-element data structure).

An example of access count improvement on various two layered dynamic data structures is shown in Table 2, whereas N is the total number of elements of the list and A is the total number of elements of the array.

Table 2. Access count example to a random element of various DDTs

Dynamic data type	Access count to a random element
CLS1WEL	$N/2A + 2$
CLS1WPLO	$N/2A + 3A/N + 1$
CLS1WPL	$N/2A + 3$
CLS2WEL	$N/4A + 2$
CLS2WPLO	$N/4A + A/5N + 5/4$
CLS2WPL	$N/4A + 3$
CLSAR	1
CLSPA	2
LL1WEL	$N/2 + 1$
LL1WPLO	$N/2 + 1/N$
LL1WPL	$N/2 + 2$
LL2WEL	$N/4 + 1$
LL2WPLO	$N/4 + 2/N + 1/4$
LL2WPL	$N/4 + 2$

In a similar fashion to 2-layered implementations, n-layered implementations of DDTs can be constructed. A common implementation of multi-layered DDTs is the “tree”. A tree is a pointer array of pointer arrays of pointer arrays etc. Each level of the tree is in reality a layer of pointer arrays, and the final level is where the records actually reside. In this paper, only single and two layered implementations will be considered, but this work can be easily extended to multilayered implementations.

3.2 Application Behaviors

Four different kinds of application dynamic behavior exist [8]. Namely, look up, iterate, insert and remove. Look up behavior corresponds to the retrieval of a data element, given a specific key value (e.g. source or destination IP address). Iteration behavior corresponds to the traversal of all the data elements that are stored in the DDT regardless of the associated key value of every data element. Insert behavior corresponds to the insertion of data elements. And finally, remove behavior corresponds to the removal of data elements.

Depending on the software design, the application under investigation may be look up dominant, iteration dominant, insert dominant, remove dominant or combinations of these. In all cases, we assume that look up, insert and remove behavior have to take place in a constant amount of time. The execution times of these operations may thus not depend on the total number of stored data element in the DDT.

Therefore the amount of total memory accesses does not depend only on the DDT selected but also on the behavior of the application. To be more precise, different behaviors favor in terms of energy consumption some DDTs more than others.

4 Dynamic Data Type Refinement Methodology

4.1 Cost Function

Wireless network applications must run on mobile devices. These devices are realized with the use of state-of-the-art embedded systems. In these embedded systems, the performance of the aforementioned applications is considered a hard constraint to meet, whereas area and energy are cost factors and must be optimized. This paper focuses on the optimization of the energy consumption factor.

The embedded systems, in which wireless network applications are realized, consist of on-chip and off-chip memories. During normal communication conditions of wireless devices [14], the main source of energy consumption is the data transfer and storage in these memories [9]. For on-chip memories, the energy consumption of one memory access increases with the memory size, while for off-chip memories, it can be considered more or less independent from the memory size, and a significant portion goes into the off-chip and communication. Hence energy can be saved either by reducing the number of memory accesses, or by storing data into smaller on-chip memories, or by doing both. Also, note that the relation between the memory accesses or the memory size and the energy consumption is not linear.

To this end, the refinement of DDTs, which make most of the data transfer and storage, should mainly aim at minimizing the memory footprint and data accesses to achieve the desired energy consumption reduction.

Finally, to estimate the consumed energy by the wireless application through the use of the various DDTs, we have used an updated version of the CACTI model [10]. This is a complete energy/delay/area model for embedded SRAMs that depends on memory footprint factors (e.g. size, internal structure or leaks) and factors originated by memory accesses (e.g. number of read or write accesses and technology used).

4.2 Refinement Methodology

As mentioned in Section 1, the dynamic memory subsystem can have important influence on the overall application performance. Our design method focuses on a high level specification of the network applications and on profiling tools that are used to extract information about the important factors concerning the memory performance, namely memory size, memory accesses and memory power consumption. As a result, repeated refinements are done and estimations from these refinements are used to define the DDT that consumes the least power. It must be noted that the functionality of the application is not changed at all during this process.

The whole process is divided in three main steps. First, the method analyzes the access pattern of the DDTs involved in the application and optimizes their implementations preserving the hard constraints set by the wireless network application. This is done with the use of the Matisse tool [4]. Secondly, global dynamic memory managers are considered in the design flow to tackle the allocation and de-allocation of the DDTs. Finally, the last step is the physical memory management which deals with the allocation of the DDTs in specific memories or memory hierarchies.

The dynamic and physical memory managers are not detailed yet, because we want to concentrate on a pure software implementation and our aim is not a specific embedded platform. Therefore, only the DDT refinement phase will be explained in the following steps:

1. **Common interface installation.** Firstly, we have to implement a certain set of basic operations about the handling of elements to provide a common interface between any DDT and the application. In this way, no matter which DDT we use, we will not have to change the application source code over and over again.
2. **Profiling tools installation.** Secondly, we are going to insert the Matisse profiling tools in every DDT, so that we can get measurements about the usage of each DDT and thus decide, which are the dominant ones that need refinement. The profiling tools can be easily embedded in any DDT with the use of a Perl script, provided with the Matisse tool. This is a semi-automatic process and can be easily applied to large programs.
3. **Search space exploration.** Thirdly, we explore the available DDT search space by implementing each DDT in the source code of the wireless network application. The memory accesses, the normalized memory footprint and energy consumption of all these DDTs are measured by the profiling tools and stored in huge log files (the normalized memory footprint is the average size occupied by each DDT). The

Matisse tool contains a C++ library of 14 single and two-layered DDTs that can be easily implemented and profiled.

4. **MATLAB calculations.** Fourthly, with the use of a minimization-problem function developed in MATLAB, we get the final values about the energy consumption of each DDT.
5. **Common interface and profiling tools removal.** Finally, we remove the “basic operations interface” and the profiling tools and implement the DDT that we have selected, which is the most energy efficient.

Due to the dynamic nature of the wireless network applications, special care must be taken in the profiling of the system. Typical trace inputs and behavior of the specific wireless network, which is designed, must be considered. For example, there are going to be differences about the most energy efficient DDT implementation between a Wireless Local Area Network (WLAN) and a Wireless Personal Area Network (WPAN). Therefore, it is common in the simulation of a wireless network, with low internet traffic, to need a very simple data type to achieve the least power consumption. On the other hand, high internet traffic wireless networks need more complex DDTs with multilayer implementations. In any case, our methodology steps must be followed to arrive in the most power efficient solution.

5 Case Studies and Simulation Results

We have applied the proposed methodology in two case studies from the network application domain: the first case study is Route [12], an algorithm that implements IPv4 routing according to RFC 1812 and the second one is DRR [13], a scheduling algorithm. The methodology can be used both to the wired and wireless domain [14][15].

In the following subsections we describe briefly the behavior of the two case studies and the proposed approach is applied to obtain profiling values (e.g. energy consumption, memory usage and memory accesses). The results have been obtained with the gcc-2.95.3 on a Pentium II at 350 MHz with 132 Mbytes SDRAM and running FreeBSD 4.8-Release 3. All the results are average values after a set of 10 simulations for each application and DDT implementation, where all the final values are very similar (variations of less than 2%). Finally, the design time needed to achieve these refinements has been one week for each application.

5.1 Method Applied to a Routing Application

The first case study presented is the Route application, which is taken from the Net-Bench benchmarking suite [11]. The application implements the table lookup along with internet checksum for the header. It makes the necessary changes to the header (e.g. to the Time-To-Live value), fragments the packet if necessary and forwards it. The source code is taken from the FreeBSD Operating System [12].

A “route” is a defined pair of addresses: a destination and a gateway. The pair indicates that if you are trying to get to this destination, communicate through this gateway. The routing table is implemented by the radix algorithm, which is a member of the Route application. Basically, the radix algorithm builds and maintains radix trees for routing lookups. A tree is being built with internal nodes and leaves. The leaves

represent address classes, and contain information common to all possible destinations in each class. As such, there will be at least one mask and prototype address. Each internal node represents a bit position to test.

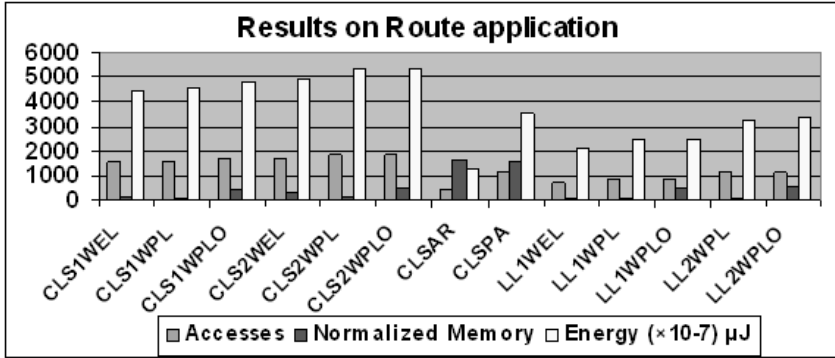


Fig. 3. Profiling values for different dynamic data types in the Route application

The data structure for the keys is a radix tree with one way branching removed. The index rn_b at an internal node n represents a bit position to be tested. The tree is arranged so that all descendants of a node n have keys whose bits all agree up to position $rn_b - 1$. There is at least one descendant which has a bit with value ‘1’ at position rn_b , and at least one with the value ‘0’ there. A route is determined by a pair of key and mask. The mask that we use has the value 255.255.255.255. The source code makes use of normal routes with short-circuiting an explicit mask and compare operation when testing whether a key satisfies a normal route, and also with remembering the unique leaf that governs a sub tree.

The routing application has been profiled in our results for an input trace of 20,000 packets. The size of the packets is varying from 0 to 512 bytes. This means that the memory needed to store the data is not fixed but it depends on the size of the incoming packet. That illustrates the dynamic nature of the Route application.

The next task to be performed is to find into the application’s source code the dominant DDTs. In our case the dominant ones are the classes `radix_node` and `rtentry`. The class `radix_node` describes the structure of the routing table, which is implemented as a radix tree, and the class `rtentry` holds the routes that have been inserted into the routing table. After the dominant DDTs have been defined, we have used the Matisse tool in order to explore the DDT search space.

Comparing the estimates provided by the Matisse tool, conclusions can be easily drawn on which DDT should be used to achieve an energy efficient implementation of the application. Specifically, concerning the routing algorithm it has been found that the DDT that leads to an optimal implementation in terms of energy consumption is CLSAR. For instance if it is required to add or delete a node to the n^{th} level of the tree, with the CLSAR implementation only one access is needed to get to the desired position, whereas using the single list implementation we should traverse the tree down to the n^{th} level and then perform the desired action. Although this DDT imple-

mentation (CLSAR) gives a 53.4% increase in terms of normalized memory footprint, it also gives a 60.9% reduction in the energy consumed and a 63.4% reduction in memory accesses in comparison to the worst solutions, the DDTs CLS2WPL and CLS2WPLO (as shown in Fig. 3). Finally, a 78.5% reduction in execution time is observed using our methodology.

5.2 Method Applied to a Scheduling Application

The second case study presented is Deficit Round Robin (DRR from now on) fair scheduling algorithm that is commonly used for bandwidth scheduling on network links [13]. The algorithm is implemented in various switches currently available (e.g., Cisco 12000 series) plus it is part of the Netbench suite [11], which consists of a set of characteristic, common networking applications. Its format categorization is queue maintenance and packet scheduling for fair resource utilization.

In the DRR algorithm, the scheduler visits each internal non-empty queue, increments the variable deficit by the value quantum and determines the number of bytes in the packet at the head of the queue. If the variable deficit is less than the size of the packet at the head of the queue, then the scheduler moves on to service the next queue. If the size of the packet at the head of the queue is less than or equal to the variable deficit, then the variable deficit is reduced by the number of bytes in the packet and the packet is transmitted on the output port. The scheduler continues this process, starting from the first queue each time a packet is transmitted. If a queue has no more packets, it is destroyed. The arriving packets are queued to the appropriate node dynamically allocating memory for them and if no such exists then it is created.

In the simulation of the algorithm, 5,000 packets are served with sizes varying from 1,500 to 15,000 bytes, while ten source and destination addresses are used. The reason of these limitations is the memory limitations of the system on which the simulation run. Still they have no effect on the proposed methodology. It should be also noted that only the size of the packet data is of importance as the algorithm does not process the data themselves.

Two dominant DDTs are presented in DRR: the first is the class `Packets` (used to create the packets to be scheduled in queues), while the second is the class `Deficit_node` (used to create the queues in which the packets are scheduled). Even between these two it can be identified that the most important DDT is holding the packets of each queue since there is one `Packets` data type for every queue, while there is only one queue data type throughout the program. Furthermore, the elements of the `Packets` can have a size varying from 1,500 to 15,000 bytes (depending on the packet size), while on the other hand each element of the queue DDT has a 16 bytes size.

Comparing the estimates provided by the Matisse tool, conclusions can be drawn on which DDT should be used to achieve an energy efficient implementation of the application. Specifically, concerning the DRR algorithm it has been found that the DDT that leads to a minimal energy consumption is CLSAR. This type of dynamic data gives a 44.3% reduction in the energy consumed (as shown in Fig. 4) and a 69.3% reduction in execution time, in comparison to the most common implementation, which is a single linked list.

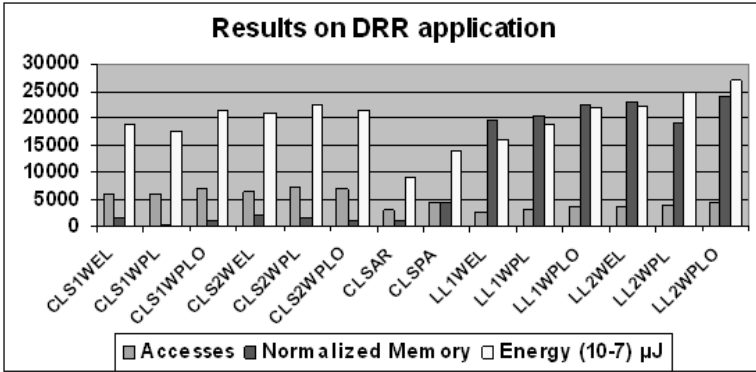


Fig. 4. Profiling values for different dynamic data types in the Route application

In order to give a notion of the spectrum of choices it should be mentioned that the worst solution concerning energy, consumes three times the amount of energy used by the optimal solution, making clear that the dynamic memory subsystem cannot be ignored when designing with low energy constraints in mind

In retrospect, we can see why an array serves our simulation better than the single linked list, a choice not so obvious for the programmer, initially. First of all, when trying to add a queue or a packet to a queue, the application does not have to traverse all the elements until it finds a null pointer (as it did with the linked list), to decide where to add the new element. Instead it can add it directly to the end of the array. The gains are even larger, when the look-up procedure does not examine the nodes one by one sequentially, but accesses to random nodes are made during runtime, because in this case the array only needs one access to get to the node, while the list needs n accesses to get to the n^{th} node.

6 Conclusions

As wireless networks grow in size and complexity, more complex wireless network applications with big dynamic memory requirements are employed to support them. In this paper, we prove the effectiveness of the proposed DDT refinement methodology to optimize the dynamic memory subsystem for the aforementioned applications.

This methodology allows a structured analysis and profiling of the memory access patterns hidden in algorithms with complex dynamic memory use. The way in which the data is stored for the studied algorithms is optimized with the use of refined DDTs. By doing this, the access patterns are also transformed and optimized, thus reducing the energy consumption of the wireless network application. Finally, the design is done in an easy step-wise method that gives the flexibility to the designer to meet the hard constraints of the application's performance and take into account the application's memory footprint as well.

Acknowledgements. Many people have contributed to the elaboration of this methodology and the development of the Matisse tool. We would like to especially thank Marc Leeman and Chantal Ykman-Couvreur from IMEC, Leuven. This work is partially supported by the European founded program AMDREL IST-2001-34379 and the Spanish Government Research Grant TIC2002/0750.

References

- [1] K. Keutzer et al. "Mescal Project", <http://www.gigascale.org/mescal/index.html>, 2002
- [2] S.Wuytack, F.Catthoor, H. De Man, "Transforming Set Data Types to Power Optimal Data Structures", *Proc. IEEE Intl. Workshop on Low Power Design, Laguna Beach CA*, 1995
- [3] C.Ykman-Couvreur, J.Lambrecht, D.Verkest, F.Catthoor, H.De Man, "Exploration and Synthesis of Dynamic Data Sets in Telecom Network Applications", *Proc. 12th ACM/IEEE Intl. Symp. on System-Level Synthesis (ISSS), San Jose CA*, pp.125–130, December 1999
- [4] M. Leeman, D. Atienza, C. Ykman-Couvreur, F. Catthoor, J. M. Mendias, "Methodology for Refinement and Optimization of Dynamic Memory Management for Embedded Systems in Multimedia Applications", *IEEE Intl Workshop on Signal Processing Systems*, 2003
- [5] L. Benini et al. "System level power optimization techniques and tools", in *ACM Transaction on Design Automation for Embedded Systems (TODAES)*, April 2000
- [6] S. Mamagkakis, M. Dasygenis, D. Soudris, and C. Goutis, "Data types, control and data flow structures of telecom network applications", *EASY Project IST-2000-30093*, February 2002
- [7] Wilson, Johnstone et al. "Dynamic Storage Allocation, A survey and critical review", *Internation Workshop on Memory Management, Kincross, Scotland, UK*, 1995
- [8] E. G. Daylight, T. Fermentel, C. Yckman-Couvreur, F. Catthoor, "Incorporating Energy Efficient Data Structures into Modular Software Implementations for Internet Based Embedded Systems", *ACM Intl. Workshop on Software and Performance*, 2002
- [9] F. Catthoor, S. Wuytack et al., "Custom Memory Management Methodology – Exploration of Memory Organization for Multimedia System Design", *Kluwer Academic Publishers*, 1998
- [10] N. Jouppi, "CACTI Model", <http://research.compaq.com/wrl/people/jouppi/CACTI.html>
- [11] G. Memik, B. Mangione-Smith, and W. Hu, "Netbench: A benchmarking suite for network processors", *CARES Technical Report*, 2001
- [12] The FreeBSD Project, "FreeBSD Operating System", <http://www.freebsf.org>, 2003
- [13] M. Shreedhar and G. Varghese, "Efficient fair queuing using deficit round robin", in *Intl. Proceedings of SIGCOMM*, Cambridge, MA, September 1995.
- [14] P.J.M. Havinga, G.J.M. Smit, and M. Bos, "Energy efficient adaptive wireless network design", *Proc. 5 th Symposium on Computers & Communications (ISCC00)*, Antibes, France, July 3–7, 2000.
- [15] K. Aida, A. Takefusa et al., "Performance Evaluation Model for Scheduling in a Global Computing System", *Intl Journal of High Performance Applications*, Vol 14 (No3), 2000.
- [16] G. Dimitroulakos, A. Milidonis, M. Galanis, G. Theodoridis, C. Goutis, F. Catthoor, "Power Aware Data Type Refinement On The Hiperlan/2", *International Conference on Electronics, Circuits and Systems*, United Arab Emirates, 2003

Context-Aware Group Communication in Mobile Ad-Hoc Networks

Dario Bottazzi, Antonio Corradi, and Rebecca Montanari

Dipartimento di Elettronica, Informatica e Sistemistica - University of Bologna
Viale Risorgimento, 2 - 40136 Bologna - ITALY
Phone: +39-051-2093001; Fax: +39-051-2093073
{dbottazzi, acorradi, rmontanari}@deis.unibo.it

Abstract. The widespread availability of both fixed and wireless network connectivity and the growing market of portable devices are enabling anytime and anywhere *impromptu* collaboration. The emergence of Mobile Ad-Hoc Networks (MANET) further opens up new possibilities for the provisioning of advanced collaborative services, such as civil protection, e-care, and troop car management. However, the design and the deployment of collaborative applications in MANET scenarios raises new group management challenges. In particular, MANET characteristics, e.g., unpredictable and frequent mobility of users/devices, intermittent device connectivity, continuous variations of network topology, make it impossible any a-priori knowledge about group members availability and ask for novel solutions to handle properly the communication about group members. The paper proposes a context-aware communication model to govern communication on the basis of the characteristics of the communicating parties, such as their location and their profiling information. The model provides communication patterns with different semantics to address both point-to-point and point-to-multipoint communication needs. The paper shows the implementation of the proposed model in the AGAPE framework for the design, deployment, and support of collaborative applications in MANET environments and presents the functioning of the AGAPE communication support in the context of a civil protection application scenario.

1 Introduction

The widespread availability of wireless network connectivity in the environments where users live and work, the increasing diffusion of portable devices, and the emergence of novel kinds of networks, such as Mobile Ad-Hoc Networks (MANET) create novel opportunities for applications that require *impromptu collaboration* between unknown partners sharing common interests.

However, the provisioning of collaborative services, such as emergency rescue, in environments with no fixed-network infrastructure and with constantly changing operating conditions raises new challenges and makes it necessary to re-think and to re-design traditional group management solutions [1]. Neither networks topology is pre-defined and fixed nor a-priori assumptions on the status and availability of group

members are possible. Group members appear and disappear in an unpredictable manner and frequently change their location and their access point of attachment to collaborative services; disconnection and network partitioning are common events. The scarce bandwidth provided by wireless network technologies makes the congestion become a normal and frequent condition. In this scenario, collaboration among users is inherently transient, it occurs among continuously varying and previously unknown partners.

Novel support solution are required to address the different group management issues that arise in the provisioning of collaborative applications. Communication is a crucial aspect that recent research activities start to address along two main research directions: uncoupled versus coupled message-oriented communication models [2], [3]. We herein focus on message-oriented communication in collaborative applications deployed in MANET environments.

Traditional solutions cannot support group communication in the new computing scenario because it is impossible in MANET environments to rely on central naming solutions and to reach a global agreement on unique names among different members. The paper proposes to exploit visibility of context information, such as the physical position of users/devices, the preferences/characteristics of group members and the status of network operating conditions, to maintain and organize group members views and to enable effective group communication. In more details, we claim that context-awareness allows to identify a communicating party on the basis of its location and of its characteristics rather than simply depending on its name.

The paper presents the implementation of these concepts in the AGAPE (Allocation and Group Aware Pervasive Environment) middleware for the support of group membership management in MANET scenarios. In particular, the AGAPE communication solution provides several context-aware communication patterns, ranging from point-to-point to point-to-multipoint ones. An AGAPE group member can decide to communicate with one specific co-located member with a specific profile, or with a group member dynamically selected among a set of members with equivalent profiles or with multiple group members with desired characteristics. No predefined knowledge on group member names is necessary in AGAPE to enable communication.

The rest of the paper is organized as follows. Section 2 presents the new communication requirements in MANET environments, Section 3 describes the AGAPE framework, and Section 4 shows the applicability of the AGAPE communication solution in the context of a civil protection application scenario. Finally, concluding remarks follow.

2 Communication Requirements in MANET Environments

Collaboration in MANET environments calls for novel communication solutions ranging from point-to-point and point-to-multipoint ones. Few different approaches are starting to emerge that address communication for MANET scenarios [2], [3], [4]. Systems such as [2], [4] propose an uncoupled communication model that relies on shared tuple spaces. Tuple spaces provide shared dataspace to put and retrieve infor-

mation in an uncoupled way, by exploiting a pattern matching mechanism. That assumes the sender does know neither the receiver, nor when the information will be retrieved. Moreover, the receiver agent can retrieve information even with a partial knowledge of it. This is particularly useful in wide and dynamic environments where a complete and updated knowledge may be difficult or even impossible to achieve and where the sender and the receiver do not interact tightly. Systems such as [4] promote message-oriented communication styles. The sender identifies the destination of messages on the basis of their names and message delivery is restricted to destination entities. Collaborative applications that require spatially and temporal coupled communication can benefit from message-oriented communication models. This paper focuses on this kind of applications.

However, the development of collaborative applications based on message-oriented mechanisms require to address several other issues. Message recipients cannot be selected and addressed on the basis of their name attribute. Group member mobility makes it difficult to rely on a-priori knowledge about names, allocations and characteristics of possibly interoperating partners, thus making inappropriate traditional naming mechanisms. The exploitation of name for communicating with a group member requires complex location tracking mechanisms. But even if available, a group member name may be un-informative or insufficiently trustworthy in MANET environments where it is impossible to guarantee name identity uniqueness. Few recent solutions are starting to emerge that provide application designer with the possibility to deliver messages on the basis of recipient characteristics and not on recipient identity [5].

We claim that the development of MANET collaborative applications may benefit from context-aware message-oriented communication solutions: the selection of message recipients should depend on the applicable context and its dynamic evaluation. Different definitions of context have been recently proposed [6], [7]. In the following, we use context as the collection of any information useful to characterize the runtime situation of a communicating party during her service session, e.g., its location, its profile and its desired collaboration preferences. For instance, the location and the reciprocal position of the different interoperating parties is a key parameter to take into account into the design of communication solutions for collaborative applications.

In particular, MANET environments suggest to promote interoperation between co-located partners. In fact, several collaborative applications, such as civil protection, require tight collaboration between neighbors [8] and to enable communication among close members connected by short-length routing paths permits to save bandwidth and to improve system robustness. The impossibility to rely on stable network connections and to achieve acceptable error rates in message delivery through long-length routing paths makes it also technically difficult to enable collaboration between distant partners.

In addition to location attributes also profiles play a key role in controlling communication. There are cases that require to select message recipients on the basis of profile attributes. That is the case of a group member interested to know whether there is another member capable of providing specific functionality. The main advantage derives from the fact that a communicating group member becomes unavailable due to

unannounced disconnection/re-connection, a potentially new one with equivalent characteristics could continue communication.

To introduce the communication problem, let us consider a civil protection service that allows different users with different profiles, e.g., physicians, firefighters, engineers, and connected via lap-top computers to constitute a MANET network and to interoperate by exchanging SMS-like messages. The service should permit civil protection operators to interoperate with a specific co-located colleague, such as their boss, in order to solve a problem they have encountered during on-site aid operations. In addition, the service should allow fire fighters to alert one co-located physician to provide initial cares to an injured man. In this case, all co-located physicians are equivalent and messages should be delivered to a randomly chosen and close one regardless to her identity. Fire fighters should be capable also of communicating a warning message to a set of co-located colleagues.

This scenario exemplifies some of the possible and different communication patterns that can be needed among group members in MANET environments. We identify and propose three different communication patterns:

- **uni-cast** point-to-point communication. When one group member has to communicate, one and only one target member is selected. The chosen target is co-located and matches specific collaboration preferences. The pattern ensures that all messages are delivered to the designated entity as long as it is reachable. As a consequence, uni-cast permits to implement long lasting (possibly) stateful sessions of interoperation between two collaborating partners;
- **any-cast** point-point communication pattern. The pattern delivers messages toward a randomly chosen co-located entity that matches a specified profile. The pattern is suited for all situations where short lasting stateless collaboration activities are needed;
- **multi-cast** point-to-multipoint communication pattern. The pattern permits to deliver the same message to all the co-located entities matching the desired profile. Similarly to the any-cast pattern, multi-cast support short lasting stateless collaboration activities.

The choice of the proper communication strategy depends on a number of application specific factors, typically related to the execution context and to the characteristics of the service.

All aforementioned communication patterns require appropriate support solutions to properly manage binding between communication parties and to re-qualify obtained bindings at run-time depending on the dynamic execution conditions. There are two main binding requirements. The uni-cast pattern requires the possibility to obtain the reference to the needed co-located member and to maintain it until it is reachable. On the contrary, in the case of any-cast and multi-cast patterns the obtained references do not have to be maintained during communication, but change dynamically to refer to different group members at any message exchange. The only constraint is that the new binding should refer to a new group member in the same locality with equivalent properties.

3 The AGAPE Framework

AGAPE supports the rapid design, development and deployment of context-aware interoperation among users operating via mobile terminals in MANET environments. Collaboration in AGAPE is based on the metaphor of group where the interoperation is restricted to the entities that are member of the same group. Each group is characterized by a group unique identifier (GID) and by a profile that specifies interests, preferences, activities and goals that should be commonly agreed by all members of its. The set of members that compose a group cannot be determined a-priori. On the one hand, new members may join/leave the group at run-time depending on application-specific requirements, and, on the other hand, user terminal mobility may cause unannounced group partitions/merges.

The notion of locality is central in AGAPE to support group management. A locality is defined as the set of AGAPE entities that are placed at a reciprocal distance lower than a determined threshold value. This value is expressed in network hops and determines the maximum dimension of a location. In particular, the entities in one locality may belong to the same or to different logical groups. An AGAPE group may be partitioned into disjoint sub-sets, called clusters, that assemble co-located group members. Note that several clusters belonging to different groups may co-exist into one locality.

AGAPE identifies two different kinds of group member entities in each cluster: cluster head (CH) entities and managed entities (ME). CH is a dynamically designated management entity which is in charge of performing group management activities within the scope of the associated cluster. For example, it is in charge of admitting/rejecting join requests and of providing managed entity a perception of the group, i.e. a view, that is limited to the scope of the actual cluster. ME exploits CH group management functionality to interoperate with co-located ME. Each group member, both CH and ME, is characterized by a personal identifier (PID) which is assumed to be unique within the group [11].

3.1 The AGAPE Middleware

The AGAPE middleware provides the needed services to support group management operations in MANET environments. We herein focus on the description of the main distinctive AGAPE services to support binding and communication. Figure 1 shows the most important services grouped in two logical layers. The group layers propagates the visibility of the group members within a cluster up to the application level and provides the services necessary to enable the formation and the maintenance of groups. The communication layers provides collaborative applications with asynchronous, unreliable message-oriented communication primitives that implement uni-cast, any-cast, and multi-cast communication patterns.

The **Proximity Service** (PS) provides discovery facilities and permits AGAPE members—both CHs and MEs—to advertise their on-line availability. At regular times

member entities broadcast a beacon to all the direct neighbours. The beacon includes both GID and PID of the sender along with the role played into the cluster (either CH or ME). In addition, a Time to Live (TTL) field that expresses the number of hops the message is to be propagated is included. Upon reception, the receiver decrements the beacon's TTL and (if non-zero) retransmits the message to all its direct neighbours with a probability $p(n)$ which decreases if the number of entities located in proximity increases. This approach permits to avoid broad-cast storming. Only group members advertise their availability but, according to the TTL field, each AGAPE entity—both members and not members—are supposed to retransmit beacons. This service does not relies on existing discovery solutions, such as Jini and UPnP, because their implementation does not address well MANET settings.

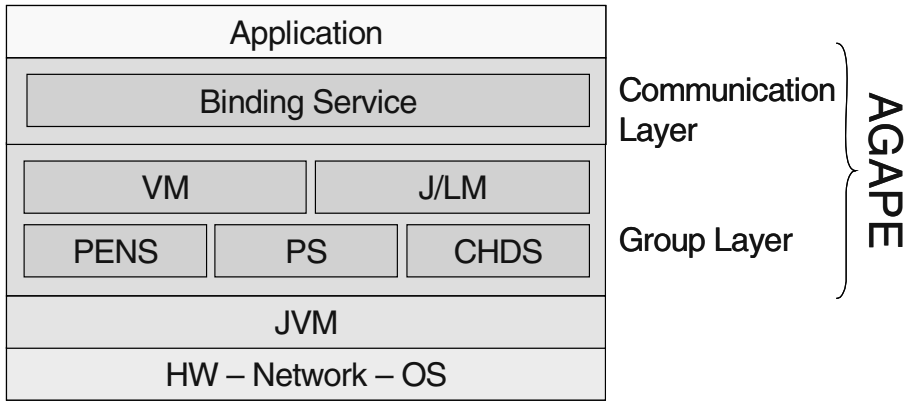


Fig. 1. The AGAPE architecture.

The **Proximity Enabled Naming System** (PENS) is in charge of randomly generating GIDs and PIDs. In particular, similarly to the naming solutions proposed for P2P environments [11], only statistical identifier uniqueness is provided. In addition, PENS maintains an updated table of co-located AGAPE entities that possibly belong to different groups and group clusters. The service senses incoming packets broadcasted by the proximity service and, according to the received information, builds a table which associates members GID/PID and role (CH/ME) with their IP address. Whenever packets from a new member entity are detected, a new entry into the table is generated. Analogously, if the reception of packets from a given member included into the table exceed a determined delay, the associated entity is removed from the table. Any change in table information is represented in terms of an event and notified to interested entities.

The **View Manager** (VM) service is in charge of creating, disseminating group views to AGAPE group members. Each group member receives a view—called Context View—that contains the list of only group member located within the scope of a cluster. In particular, each Context View entry associates each member reference with user/device profile information—such as GID, PID, user interests, age—that are necessary during the process of message recipient selection. When group members con-

nect or disconnect from the network, AGAPE reports the view changes to all interested group members into the cluster. In addition, VM customizes the context views delivered to group members on the basis of their collaboration preferences. Delivered Context Views contain only the locally available group members that match the collaboration preferences expressed by a group member at its group joining phase.

The **Join/Leave Manager** (J/LM) service allows not member entities to join the group and to member entities to leave it. By exploiting the PENS, locally available groups can be discovered. In particular, the CH entities are discovered and their visibility propagated at the application level. The joining phase consists in first contacting all discovered CHs and ask for their group profile. If a group of interest is retrieved a join request is sent to the CH in charge of group management. The join request message includes candidate member profile along with its profile describing its characteristics and the desired collaboration preferences. According to the installed management policies CH decides whether to accept or to reject the join request. In the case of acceptance an acknowledge message is replied that includes the GID/PID generated by PENS for the new member. Finally, the Join-Leave Manager coordinates with the View Manager in order to distribute to the group member the Context View related to the cluster of interest. When a member leaves a group the Join-Leave Manager coordinates with PENS to delete the GID/PID of the member willing to leave and with the View Manager to update the Context View.

The **Cluster Head Designation Service** (CHDS) is in charge of electing a new cluster head. A variation of the election protocol proposed for MANET in [9] is exploited to designate the new CH entity. The election is triggered either by the unavailability of the Cluster Head or by its inability to continue to carry on its management duties due to events, such as battery degradation or the decreasing of free memory.

The **Binding Service** (BS) supports message-oriented communication by managing on behalf of group members the bindings with their communicating parties. BS provides different binding management strategies depending on the communication patterns exploited. In particular, when a group member wants to establish a communication, it contacts BS, provides it with a Searching Profile (SP) specifying its collaboration preferences and specifies the desired communication pattern. BS exploits the SP to filter the content of the locally available Context View in order to retrieve the list of all co-located group members that match specified profiles along with their addresses (target members set – TMS). Then, BS builds a record including various fields: the GID/PID of the requesting group member, the SP, the TMS and the desired communication pattern. Each record is stored into the Binding Table directly managed by BS. Finally, BS returns to the requesting group member an handler to the record in the Binding Table. If the uni-cast communication pattern is selected, BS binds the requesting group member with the first group member in the TMS as long as it is reachable. In the case of any-cast and multi-cast communication patterns, BS binds at each message exchange the requesting group member with the first available group member in the TMS.

In addition, BS constantly keeps the TMS in the Binding Table records updated through the coordination with the View Manager service. In particular, if a relevant change in the Context View occurs, the TMS field of the Binding Table records is updated accordingly. This allows to dynamically re-qualify bindings when needed.

4 Case Study

To illustrate the functioning of the AGAPE communication support let us consider a simplified civil protection application scenario that allows impromptu interoperation between co-located operators in case of major disaster. In this scenario, the impossibility to make any assumption about the availability of network connectivity through Tetra, GPRS and UMTS channels suggests to exploit MANET infrastructure support. Our civil protection application prototype enables user to interoperate by exchanging SMS-like messages without the need for connectivity to the Internet, in a decentralized fashion, by exploiting the full-visibility of only locally available group members.

In our prototype, a Mobile Ad-Hoc network is dynamically constituted by exploiting 802.11b-enabled laptops provided to the different civil protection operators. The operating system we installed on lap-tops is Linux; to implement the MANET infrastructure we have configured lap-tops to include the AODV [10] routing protocol. Moreover, due to the lack of standard addressing schema for MANETs, we have statically configured device IP addresses. These deployment setting choices do not undermine the generality of the results. The AGAPE infrastructure does not depend on the availability of a specific routing protocol or addressing schema.

All AGAPE services are installed on each device and implemented on top of J2SE 1.4. As a consequence each user device may become on its turn either CH or ME.

The application aggregates together in one single group civil protection operators operating within the same area. Different users have different roles and competences and, as a consequence, they are characterized by different profiles. These profiles along with their used access terminal characteristics have been modeled as CC/PP profiles. For description simplicity, and without lack of generality, let us suppose that the profile includes only operator's name and skills (e.g. engineer, physician, and so on).

Let us show how AGAPE works by considering the case of one civil protection operator that promote the dynamic formation of the civil protection group at execution time. To this aim, the application client module of the civil operator allows him to specify the group profile—in our example “Civil Protection”—along with the user own profile—for example “Tom, Firefighter”. Then, the operator's lap-top exploits the locally installed PENS to generate GID/PID and the View Manager service to initialize Context Views. The operator's laptop advertises by means of the Proximity Service its on-line availability by sending beacon messages. All lap-tops into the locality can therefore benefit from the visibility of the novel Cluster Head. In particular, not-member entities can query the discovered CH to the purpose of obtaining the group profile. In our example, the different entities recognize that the CH is associated with the civil protection application and request it to join the associated group by sending it a “Request to Join” message. The message includes member profile along with the preferences about desired interoperating partners. For example, a fire fighter may require to interoperate only with fire fighters and physicians. Then, if the CH admits the new member, the Group ID along with the Personal ID that identifies the entity within the group, are delivered to the new member. In addition, the CH updates the Context Views and installs them on the new member. Note that the installed Context View reflect the preferences made at the join request. In the described example

the Context View delivered to the fire fighter device will include only fire fighters and physician members.

Let us now consider how AGAPE takes care of communication in the case an any-cast communication pattern is selected, for instance when a fire fighter wants to alert one physician—regardless to her identity—to asks for her intervention. To communicate the alert message the application that runs over the fire fighter’s lap-top must obtain a handler to the Binding Table. To this aim, the application specifies the required communication pattern along with the Searching Profile. In particular, the SP states that the desired members for collaboration are physicians. The Binding Service on fire fighter’s computer filters the delivered Context View to identify all members with a profile that matches the provided SP, i.e. all physicians, located into the cluster. Then, the Binding Service creates a new entry into the binding table. This entry must be referred to dynamically determine the recipient of messages to be sent. Note that the Binding Service updates the entry into the table according to the information it periodically gathers from the View Manager, thus dynamically re-qualifying the bindings.

5 Conclusions and Ongoing Work

The design, development and deployment of group membership and communication systems in MANET environments raise challenging issues. As a consequence it is necessary to re-think and re-design traditional group management solutions. AGAPE intends to give a contribution to the research area of middlewares to support group-aware applications in MANET environments.

AGAPE supports collaboration through the metaphor of group. Only group members entities are enabled to collaborate together. As a key feature, AGAPE implement different communication patterns that exploit the visibility of the location of group members along with their profile as a first-class concept to select and refer to desired communicating entities.

For the sake of description simplicity of AGAPE functioning, we have presented a simple civil protection application prototype built on top of AGAPE, but we are experimenting the AGAPE middleware in a wide variety of scenarios. First experiences in the use of AGAPE have shown that our middleware can simplify the design and implementation of collaborative services. These results are stimulating further research along different guidelines to improve the current prototype and to develop more complex services on top of it. We are currently working on evaluating alternative group member management models. In particular, we are investigating the possibility to rely on fully decentralised management solutions with no local central point of management, i.e., without a CH in each locality. We are also testing different algorithms for optimizing bandwidth usage of the probabilistic flooding protocol implemented by the Proximity Service. In addition, we are investigating the security concerns that group management arises by starting to integrate initial security support services in AGAPE.

Acknowledgements. This investigation is supported by MIUR within the framework of the FIRB Project "WEB-MINDS" and by CNR within the framework of the Strategic Project "IS-MANET".

References

1. D.Powell (ed.) "Group Communication", Special Issue on Group Communications, Communications of the ACM, ACM Press, Vol. 49, No.4, April, 1996.
2. G.Picco, et. al., "Lime: Linda meets Mobility", Proc. 21st Int. Conf. on Software Engineering ICSE1999, CA, May 1999.
3. L. Capra, et. al., "XMIDDLE: A Data-Sharing Middleware for Mobile Computing", Personal and Wireless Communications Journal, Kluwer, Vol. 21, No.1, April 2002.
4. G. Kortuem, et. al., "When Peer-to-Peer comes Face-to-Face: Collaborative Peer-to-Peer Computing in Mobile Ad-hoc Networks", Proc. International Conference on Peer-to-Peer Computing (P2P2001), Sweden, Aug 2001.
5. M.A.Munoz, et. al., "Context-Aware Mobile Communication in Hospitals", Special Issue on Handheld Computing, IEEE Press, Vol. 36, No.9, September, 2003.
6. A.K. Dey, et. al., "Towards a Better Understanding of Context and Context-Awareness", Proc. of Conference on Human Factors in Computing Systems CHI2000, Netherlands, April 2000.
7. T. Rodden, et. al., "Exploiting Context in HCI Design for Mobile Systems", Proc. of Workshop on Human Computer Interaction with Mobile Devices, Scotland, May 1998.
8. G. Kortuem, et. al., "Wearable Communities: Building Social Networks with Wearable Computers", IEEE Pervasive Computing Magazine, Special Issue on Wearable Computing Technologies & Applications, October-December 2002.
9. N. Malpani, et. al., "Leader Election Algorithms for Mobile Ad Hoc Networks", Proc. 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications, 2000.
10. C.E. Perkins, et. al., "Ad hoc On-Demand Distance Vector Routing." Proc. of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, LA, February 1999.
11. A. Rowstron, et. al, "Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems". Proc. IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Germany, November, 2001.

An Empirical Wi-Fi Based Location Mechanism for Urban Search and Rescue Operations*

Roberto Aldunate¹, Miguel Nussbaum¹, and Feniosky Pena-Mora²

¹Computer Science Department, Catholic University of Chile,
Santiago, Chile
{raldunat, mn}@ing.puc.cl

²Civil and Environmental Engineering Department,
University of Illinois at Urbana-Champaign
Champaign, IL, USA
feniosky@uiuc.edu

Abstract. Locating trapped people under rubble is a key task that first responders must develop as efficiently as possible. Inhospitable and chaotic environments resulting from a natural or human-induced disaster make inappropriate the use of the Global Positioning System, or other semi-infrastructure solution, for localizing survivors due to its limited outdoor operation or low probability of use in ordinary life. This article presents a simple method for locating peers considering IEEE802.11b enabled devices. The method considers the variations produced in Signal Strength between IEEE802.11b enabled devices when the body of a person interrupts their line of sight. As results, empirical models were obtained for distance and angle measurements. Using those models, the error for distance measurements was at most 20m for distances between 1 to 100m, and the error for angle measurements was at most 18° for distances between 1 to 80m, and most 38° for distances between 80 to 100m. In addition those results are used to fundament an extension to a probabilistic detection model.

1 Introduction

Urban search and rescue operations conducted after the occurrence of natural or human-induced disasters, like earthquakes or terrorist attacks, must satisfy requirements and overpass constraints that the resulting inhospitable and chaotic environment imposes. One of the most relevant requirements is that search and rescue must be developed efficiently due to the higher probability to find trapped people alive in the first hours than in the subsequent hours after the occurrence of the extreme event [14]. Among the constraints characterizing such environment are rubble, outdoor/indoor operations, and collapsed communication and energy infrastructures. In recent years, the idea of use mobile short-range wireless technology, usually addressed under the

* This research was supported in part by the Chilean National Commission of Scientific and Technological Research under Grant FONDECYT 1020734, and Microsoft Research.

concepts Mobile Ad Hoc Network (MANET) and Sensor Network (SN), has been calling the attention of researchers due to its inherent infrastructure-less characteristic. Although requirements like energy-efficient algorithm, scalability, performance, usability, and availability using limited power and resources impose additional constraints to MANETs and SNs, these are envisioned to support and provide localization, communication and coordination services for first responders.

Location systems for mobile-computing applications can be classified in terms of their dependence on some infrastructure and or if their physical measurements are absolute or relative [1]. On one hand, among the infrastructure-dependent location systems are the Global Positioning System (GPS) [2], RADAR [3], Wireless Andrew [4], and Active Badges [5]. GPS, maybe the most popular location-sensing system, is a Radio Frequency (RF) based system that provides the absolute outdoor physical position of any object on earth's surface by triangulation, using 24 geo-stationary satellites. RADAR is an in-building RF based user location and tracking system which uses which uses base stations to determine absolute indoor location of mobile IEEE802.11b enabled devices carried by users with respect to the floor layout. Wireless Andrew, which also uses IEEE802.11b technology, is a system oriented to provide not only indoor, as RADAR does, but also outdoor localization for mobile users based on a fixed infrastructure in a university campus environment. Finally, Active Badges is an indoor badge sensing system where every person located by the system wears a small infrared badge in an indoor environment. On the other hand, among the infrastructure-less solutions is Spot On [6], which uses radio signal attenuation to estimate distances among devices wearing a no off-the-shelf Spot On tag. Spot On, as well as GPS and Active Badges, uses wireless technology specifically developed for location purposes. A meaningful difference with systems like RADAR and Wireless Andrew is that use wireless technology which allows not only to be used for locating purposes, but also as the communication medium among devices. Such characteristic, as mentioned, has been one of the most important elements to consider in the development of the work presented in this article.

In some hostile environments, like search and rescue missions in disaster relief environments, a person needs to locate peers or devices respect to her own reference frame, due to the eventual unreliable situation of any fixed communicational infrastructure. What makes different our work from the ones found in literature is that we developed a system completely infrastructure-less, like Spot On, but using the broadly adopted IEEE 802.11b (Wi-Fi) technology for handheld devices as well as for cell phones, which is attractive for two reasons. First, it is more accessible for first responders, and other associate users or common people. Second, reduces the operational problem of having people using special hardware tags permanently in ordinary life due to the low-probability nature of extreme events []. In terms of localization, in this work the positions are relative and are not estimated using triangulation since it requires knowing the positions of at least three nodes for 2-D resolution [7], but using polar coordinate estimation. In terms of precision the goal is not to achieve GPS precision but an adequate for first responders to find trapped survivors, i.e., a precision of about 15 m [2]. When this precision is sufficient, it is possible to avoid both extra cost and additional power consumption associated to the use of the GPS system, as well as limitations imposed by semi-infrastructure solutions.

The main control variable used in this research effort is the interference produced on Signal Strength Quality (SSQ, $\in [0, 100]$) by the person wearing or carrying the Wi-Fi device on hands. More precisely, the interference produced on SSQ between two devices, one used by the person and other which is at a distance D , when they haven't line of sight. To the best of our knowledge, no previous research efforts has characterized impact of human-body on MANET's applications, and used it to determine location of Wi-Fi enabled peer devices. The main idea is that when a person carrying a Wi-Fi device rotates, the SSQ received by the device will gradually change. Empirical models are obtained to determine distance and angle of a remote device monitored by a monitoring person with a Wi-Fi enabled device. Considering those results a probabilistic detection model, based on [11], is presented for the environment that motivates this research. Finally, this article is part of a research in progress oriented to advance the knowledge on how using computer tools, and particularly wireless technology, the processes carried out by a response system can be improved to reduce the impact of extreme events on urban zones in terms of affected physical infrastructure and people.

The remainder of this article is organized as follows. Section 2 describes the developed method, and implementation details. Section 3 shows the experiments carried out to gather data to build empirical models for distance and angle with respect to the observer. The results obtained appear in Section 4. Section 5 introduces an appropriate probabilistic detection model for first responders detecting Wi-Fi enabled devices under rubble.

2 Local Positioning Awareness

The method developed in this research to locate 1-hop neighbors is based on both traditional radar systems and interference problems caused by human body reported by [3]. Particularly, the differences on SSQ values whether or not the user interrupts the line of sight.

Our initial measurements allowed us to observe that when two Wi-Fi enabled Pocket PCs, carried by two stationary people respectively, are at a distance D (greater than 0) and are in a line of sight, the Radio Frequency (RF) signal strength reported by their Wi-Fi network cards is a measure S_1 . If the two people remain at the same distance D , but one of them rotates up to 180° with respect to the other peer, the RF signal strength changes to S_2 , where S_1 is greater than S_2 , as represented by regions 1 and 3 in Figure 1. Also, it was noted that region 2 in Figure 1, where the target node is in the vicinity of 90° , the SS value perceived was lower than S_1 but higher than S_2 . Consequently, it was observed that the communication range decreased from the maximum value at 0° to the minimum value at 180° , and increased again up to the maximum value when the peer completed the rotation from 180° to 360° .

Although this work has been carried out on very controlled scenarios, considering only a particular situation to develop the experiments (one mobile monitoring peer and one stationary monitored device or person holding a device), e.g., a group of people performing the location process simultaneously has not been developed and physical

obstacles have not been considered, it is envisioned by us that having this problem solved it will be easier to treat with those more complex scenarios.

Based on the initial observation described above, we envisioned that using the information about SS perceived by a Wi-Fi enabled device on hands of a person rotating on her own axis, a simple algorithm for locating a stationary Wi-Fi enabled target device can be elucidated. The technique underlying this research is the same as works traditional radar systems, so the originality of this work is not in the method itself, but in the application of such concept to develop a completely autonomous localization method for inhospitable and chaotic environments where people must be detected as soon as possible.

The algorithm to be validated by the experiments is the following:

- (a) The monitoring person starts a process of gathering SSQ in the Pocket PC at the same time that starts to rotate in clockwise sense on her own axis, at a constant speed. When the person is again in the starting position, stops the gathering SSQ process in the Wi-Fi enabled device.
- (b) Data is heuristically inspected to find out the representative maximum value of the sample, point which is assumed to be the SSQ obtained when both devices were in 0° orientation.
- (c) Once the representative maximum value is obtained, the distance is estimated using an empirical signal propagation model. To determine the angle, which is measured respect to the start position of the monitoring user, the position in the sample where a representative maximum is found with respect to the whole rotation is mapped to a $0^\circ - 360^\circ$ scale. The angle obtained is the angle of the target with respect to the initial position of the monitoring person (0°).

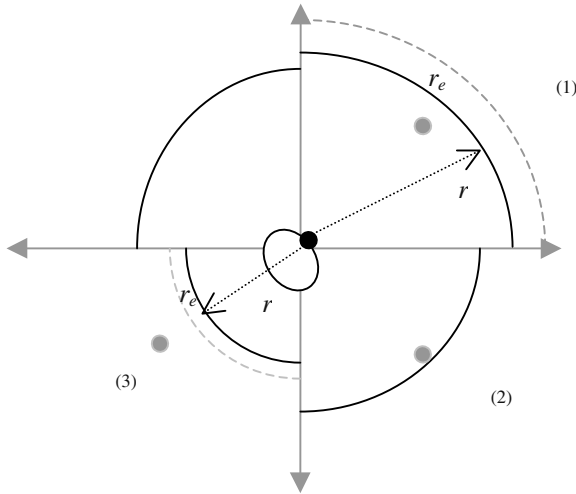


Fig. 1. Layout of different SS, between Wi-Fi enabled Pocket PCs, depending on human body interference. Region (1) represents a vicinity of 0° orientation where both devices (black and grey) have line of sight and SS can be maximum. Region (2) represents a vicinity of 90° orientation where SS is lower than in region 1. Region (3) represents a vicinity of 180° orientation where SS perceived by the monitoring device is minimal

2.1 Implementation

The instantiation of the general situation described in the previous section was implemented using Pocket PCs for both the monitoring as well as the monitored device. The monitoring device was the Compaq iPAQ 3630 Pocket PC running Linux Operating System, having an Intel ARM 206 Mhz, 32 MB in RAM, and a Cisco Aironet 340 IEEE802.11b wireless card. The Aironet driver extracts the SS from the card firmware each time a broadcast packet arrives, and it makes it available to the user-level applications through the *proc* interface. The SSQ, as a measure of quality, takes a value between 0 and 100, which is mapped from the SS value expressed in dBm. Besides to configuring the wireless card in order to establish connection with its neighbor, no other specific tuning was made, like fixing the data transfer rates or force it to transmit at maximum power, intending to use the devices in the same way that any non-specialized user would do.

In the making of the graphic user interface we used a *gcc* cross-compiler for generating ARM code, and we also used *gcc* to test the code under the Qtopia desktop environment for Linux, under X windows.

As monitored devices we used two different Pocket PCs in order to validate independence of specific hardware. On one hand, a Compaq iPAQ 3630 Pocket PC running Windows Pocket PC 2002 with 32 MB of RAM, having an Intel ARM 206 Mhz processor, and using a Compaq WL110 IEEE802.11b card. On the other hand, a Toshiba e740 Pocket PC running Windows Pocket PC 2002 with 64 MB of RAM, having an Intel ARM 400 Mhz processor and an embedded IEEE802.11b WLAN interface.

3 Experiments

In order to determine a valid control variable to be used for determining positioning we evaluated RF signal time-of-flight, packet loss rate using the ICMP protocol and RF signal strength. The results obtained showed us that only the latter was useful to be applied; resolution of time (milliseconds) returned by operating system (pings) and or not significant time of flight for 1-hop neighbors compared with time for receiving (without buffering) a packet and making it available for upper layers.

As the localization estimation underlying this work is a polar representation, the experiments were initially conducted in order to estimate independently distance and angle of the target Wi-Fi enabled device with respect to the monitoring Wi-Fi enabled device on hands of a person. The first experiments were developed to establish a signal propagation model which maps Signal Strength Quality (SSQ) to distance, then, measurements in order to determine the relation between angle and SSQ (distance is maintained invariable for these experiments) were conducted. At that time it was observed that, as will be described in Angle Measurement Sub-Section in Experiments Section, the way the person orientates the Wi-Fi enabled device with respect to her shoulders, i.e., whether it is in straight or orthogonal orientation respect to her shoulders, Figure 2 (a) and (b) respectively, has significant impact on the measurements.

Hereafter it must be understood that all data gathered was collected using orthogonal-oriented position, as illustrated in Figure 2 (b), unless the opposite is explicitly mentioned.

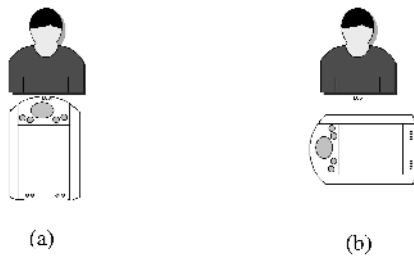


Fig. 2. Two different ways a person holds the monitoring Wi-Fi enabled device. In (a) the Pocket PC is straight-oriented with respect to the person’s shoulders. In (b) the Pocket PC is orthogonal-oriented with respect to the person's shoulders. Different SSQ curves are obtained in each case

3.1 Distance Measurement

Figure 3 shows that similar SSQ curves were obtained, which highlight that both wireless cards comply in the same manner the IEEE 802.11b standard specification, in function of the distance between the monitoring and the monitored device oriented at 0° (360°), for two different monitored Wi-Fi enabled Pocket PCs. It is clear that the more the distance, the less the SSQ. Each SSQ value present in the graph represents the mean of 2000 SSQ measures for the same distance.

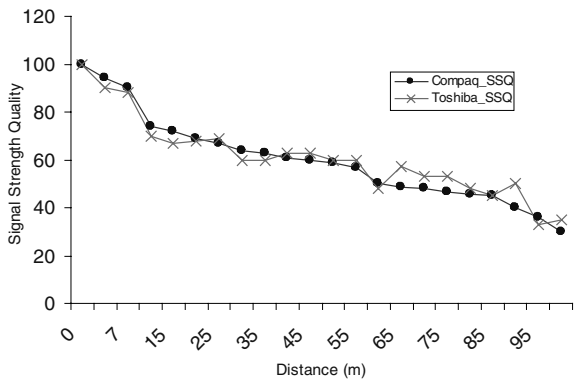


Fig. 3. Signal strength quality curves obtained in a Pocket PC equipped with a Cisco Aironet 340 WLAN oriented in 0° (360°), with respect to two target pocket PCs wearing a Compaq WL110 and an embedded Wi-Fi LAN respectively

Lower SSQ values were obtained for the corresponding same distances when the monitoring person, having the device on her hands, is oriented at 180° with respect to the monitored device, as shown in Figure 4. Another significant outcome is that similar graphs to the ones shown in Figure 3 and Figure 4 were obtained for outdoor measurements on a campus yard populated by students in movement, with a density of about 3 people for each 10 square meters.

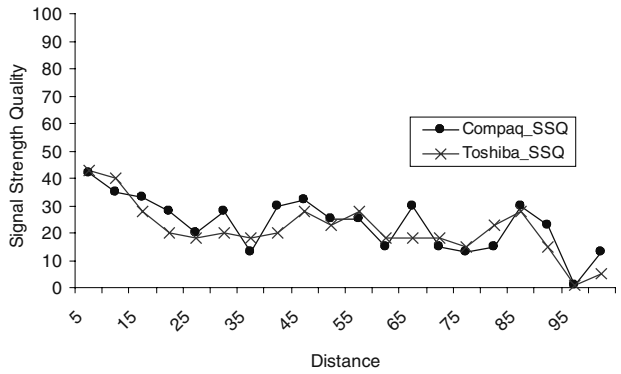


Fig. 4. Signal strength quality in function of distance for target Wi-Fi enabled Toshiba and Compaq Pocket PCs. Values obtained in a Pocket PC equipped with a Cisco_Aironet 340 WLAN oriented at 180° to target pocket PCs

The best model for the data shown in Figure 3 is the following 4th degree polynomial, where y = distance and x = SSQ:

$$y = -1.8e-005x^4 + 0.053x^3 - 0.53x^2 + 20x - 1.5e+002 \tag{1}$$

The previous model allows us to obtain a better Norm of Residuals (NR) than other curves, as shown in Table 1. In particular, 5th and 6th degree polynomials were discarded because they were too close to the raw data, without providing significant enhancement in measurements, which become unrealistic due to variances in the collected data.

Table 1. Norm of residuals for different models adjusted to data shown in Figure 3

	Models					
	Linear	Quadratic	Cubic	P.4 th	P.5 th	P.6 th
Norm of Residuals	43.909	28.334	19.038	15.711	14.548	13.588

3.2 Angle Measurement

In contrast to distance, to determine the angle a heuristic approach was used due to the lower error rate obtained compared to when mathematical models were used. We began observing the SSQ data gathered by the monitoring device when the user rotated 360° , at a constant speed in approximately 8 seconds, on his/her own axis, and the monitored device was stationary. During each rotation about 200 measures of SSQ were collected. Initially, our measures were obtained using the layout shown in Figure 2 (a) for the monitoring user. For a distance of 10m, and in a 0° orientation for both devices, i.e., each device was in front of the other, we obtained the data shown in Figure 5.

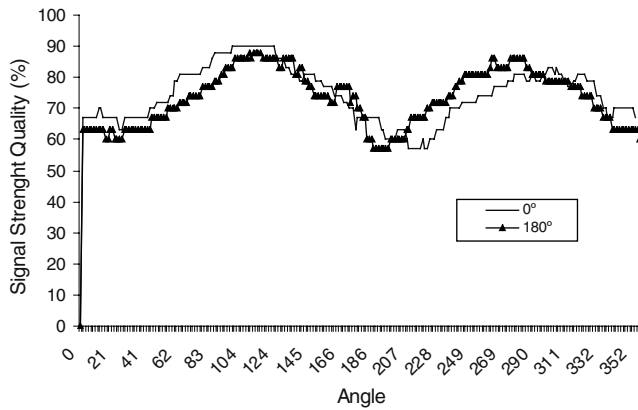


Fig. 5. Signal strength quality obtained when the monitoring Pocket PC carried in straight-oriented position by a person and she rotates on her own axis. It shows the curves for the cases when initially the monitoring is in front (0°) and back (180°) with respect to the monitored device

In opposition to our initial expectations, the maximum SSQ values were obtained at approximately 90° and 270° , which clearly did not match with reality. The reasons for such results is that the Cisco Aironet 340 card, as others, has physically two embedded antennas, known as diversity antenna, to enhance RF communication [9], and we assume they are located in the superior right and left corners of the card. We realized that when the monitoring user held the Pocket PC in orthogonal orientation and close to the user's chest, as shown in Figure 2 (b), one of the antennas was blocked, and as a result the device worked as having only one, as it is shown in Figure 6. It can be also observed in Figure 6, that the peak value for SSQ is obtained when both Wi-Fi enabled devices are oriented at 0° .

As mentioned before, to determine the peak of the curve a heuristic approach was used since the global maximum was not adequate due to the SS fluctuations. The peak SSQ value is determined dividing the curve in four sections, choosing the section with the high mean value, and selecting the axis value corresponding to the middle of the section. The divisions of the curve in four sections was selected after having checked

all possible sizes of sections from the larger to the smaller, or said in a different way; from 1 to 360 number of sections.

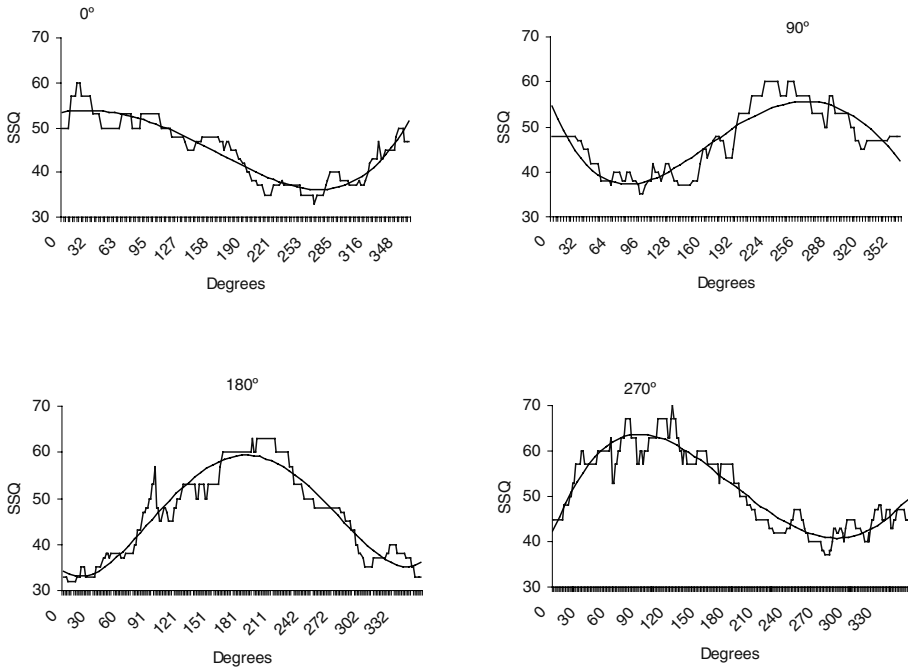


Fig. 6. Signal strength quality curves obtained for rotations when the Pocket PC is carried in orthogonal orientation. Maximum values of curves are obtained when the rotating Wi-Fi enabled device is in front (0° orientation) of target Wi-Fi enabled device

The graphs presented in Figure 6 also show that the signal strength quality value, when the person rotates at a constant speed, behaves in a sinusoidal way. In addition, although the initial diagram shown in Figure 1 is a discrete representation of such variation, the idea that SSQ will vary according the angle between the monitoring person (holding the device) and the monitored device, is confirmed. Consequently, as SSQ perceived by the monitoring device is blocked by the user the communication range is shortened.

Finally, a set of equivalent experiments were developed in an in-building context where the line of sight between devices was interrupted by both cement and light material walls, like wood or lime, to evaluate the interference of such obstructions. Although in general terms we observed that different variances were associated to different material, and its geometry, interrupting the line of sight, no significant result can be expressed. Variances in all the situations were so high that it was not possible for us to infer another behavior than a random one.

4 Results

The empirical models obtained in the previous section were applied in an outdoor environment, and gave the results shown in Figures 7 and 8. Figure 7 shows the error produced in distance measurements. Below 10 meters and above 90 meters it is very precise since SSQ is stationary high (strong) and low (weak), respectively. In contrast, the rest of the curve shows in general higher errors and higher standard deviations.

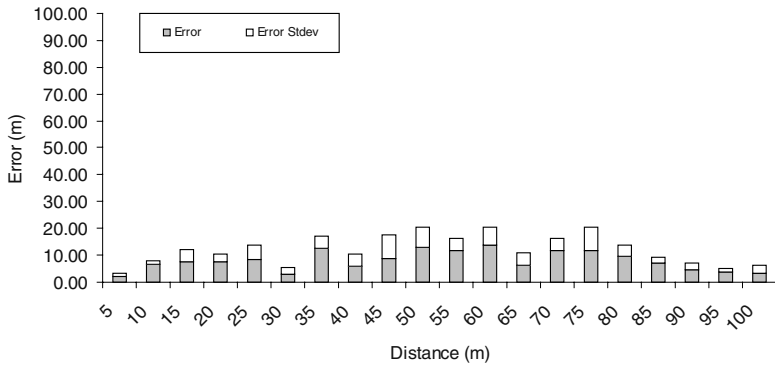


Fig. 7. Error and standard deviation associated to distance measurements using the model obtained empirically

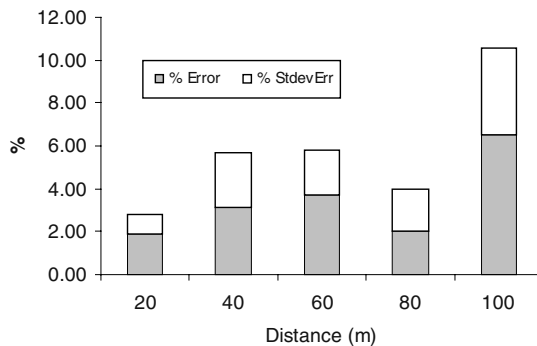


Fig. 8. Error percentage and standard deviation percentage associated, depending on distance, in measurements of angle. Up to 80m the total error percentage is at most 6 percent, above 80m error increases as the distance increases

The precision in angle measurements at different distances is shown in Figure 8. Up to approximately 80m of distance the angle error was at most 18°. Above 80m and up

to 100m the angle error was at most 38° . It is consistent with the data shown in Figures 3 and 4; SSQ measures for 0° and 180° , and consequently any orientation in between, become less differentiable for distances greater than 80m. Hence, the more the rotation curve becomes flat, the less the significance to find a SSQ maximum.

5 Probabilistic Detection Model

Real life situations like searching for trapped survivors under rubble demands an adequate detection model to improve localization techniques. As it is shown in the previous section, human-body may have strong impact on signal strength perceived by a Wi-Fi enabled device used by a first responder, making uncertain the applicability of underlying platform services like routing based on signal strength [10][13] or probabilistic detection models in such environment. Of particular interest for this research is to apply the empirical results obtained to improve theoretical models for the described environment. Therefore, a probabilistic detection model based on [11], which in turn is based on [12], is presented here. In [11] we found the following probabilistic detection model for sensor networks:

$$C_{xy}(Si) = \begin{cases} 0, & \text{if } r + r_e \leq d(Si, P) \\ e^{-\lambda a \beta}, & \text{if } r - r_e < d(Si, P) < r + r_e \\ 1, & \text{if } r - r_e \geq d(Si, P) \end{cases} \quad (2)$$

“Where r_e is a measure of the uncertainty in sensor detection, $a = d(Si, P) - (r - r_e)$, and α and β are parameters that measure detection probability when a target is at distance greater than r_e but within a distance from the sensor Different values of the parameters α and β yield different translations reflected by different detection probabilities, which can be viewed as the characteristics of various types of physical sensors” [11] (although the paper describes α , it’s easy to find out that it is a typographic error; it should be λ).

Based on the results presented in the previous section the probabilistic detection model considering a person wearing or carrying a short-range wireless enabled device should be the following:

$$C_{xy}(Si) = \begin{cases} 0, & \text{if } r(\theta) + r_e(\theta) \leq d(Si, P) \\ f(\theta)e^{-\lambda a \beta}, & \text{if } r(\theta) - r_e(\theta) < d(Si, P) < r(\theta) + r_e(\theta), \text{ and } \theta \in [0, 2\pi] \\ f(\theta), & \text{if } r(\theta) - r_e(\theta) \geq d(Si, P), \text{ and } \theta \in [0, 2\pi] \end{cases} \quad (3)$$

Where $f(\theta)$ ($\in [0, 1]$) is a function of the angle between the sensor carried by the person and the device to be detected, with $f(\theta=0) = 1$ and $f(\theta=\pi) = 0$. For the particular case of Wi-Fi, analyzing data gathered in our experiments, a representation which moderately fits the curves shown in Figure 6 is $f(\theta)=K + (1+\cos(\theta))/L$, where $L = 2/(1-K)$ and $K = \text{Min}\{\text{SSQ}/100\}$ for a complete rotation (value obtained in the vicinity of 180° orientation). In addition, As the SSQ exhibits a sinusoidal behavior depending on θ , r and r_e also depend on θ . It is envisioned by us that $f(\theta)$, as well as λ and β , is dependent on the specific technology used in each implementation, i.e., IR, RF, Ultrasound. Nevertheless, advancing knowledge about such consideration will be part of further research. The impact of the introduction $f(\theta)$ to the probabilistic detection model taken from [11] is shown in Figure 9. Instead of having one curve per device, as was the case in the previous model, there is one curve for each different angle between devices per each device, provided that a person is carrying the monitoring device. Another important conclusion obtained from the curves is that the probability that a device be detected by the one carried by a person will be limited by the angle between such devices.

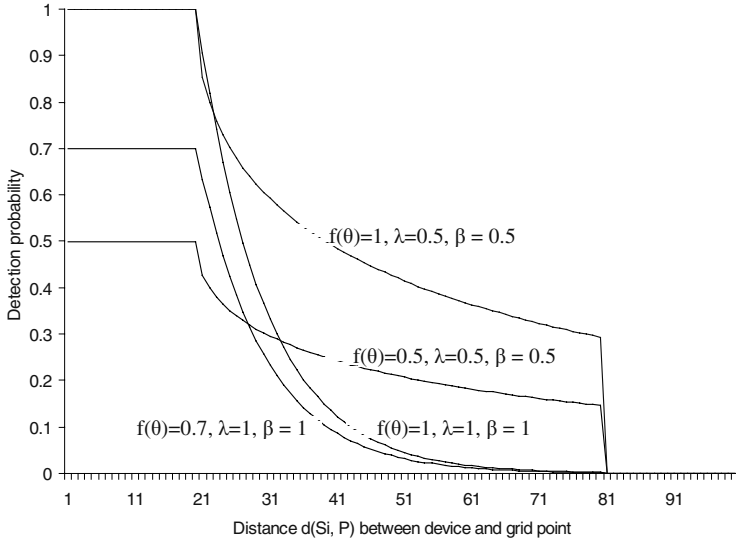


Fig. 9. Adapted probabilistic sensor detection model for situations when a person is an element of the system wearing or carrying on hands the monitoring sensor

6 Conclusions and Future Work

A simple method to support location awareness of Wi-Fi enabled devices for first responders has been empirically obtained. This method is based on the interference

that the human body produces on RF Signal Strength when the line of sight between Wi-Fi enabled Pocket PCs is interrupted or altered. On one hand, our results show that it is possible to know the distance of a monitored stationary device to a monitoring one in hands of a person with a precision of approximately 12m. On the other hand, it permits to know the angle with a possible error of at most 18° if the monitored device is up to 80m, or at most 38° if it is above it and up to 100m.

It has been demonstrated that the person's body have high impact on signal strength perceived by a Wi-Fi enabled device on hands of the person. Also, it has been demonstrated that such impact behaves in a sinusoidal way as the person rotates on its own axis and the monitored Wi-Fi enabled device remains stationary.

Application-driven development demands to consider each factor relevant in theoretical models. That's why the results obtained in this research must be considered in the development of underlying services or models like probabilistic detection models when people is a component of the system modeled. In that sense, the outcomes obtained are used to achieve a more realistic probabilistic detection model for detection of IEEE802.11b enabled devices. This is particularly useful for search and rescue operations of first responders.

Although the outcomes obtained in this research are lower than the ones provided by GPS-based systems, they are significant for applications that require only relative location to the monitoring person; like in search and rescue operations developed by first responders. As human beings we hardly can perceive differences among few degrees or meters, i.e., we usually locate objects in the world as being in our right, left, front, rear or some combination of them.

Although the application developed running on the monitoring device can only monitor one device at a time, the extension of this to a group of devices obey the same scheme. In addition, part of the future work is the consideration of a more realistic environment including obstacles, and the consideration of real 3-D world. Finally, current GPS-free ad hoc positioning systems can take advantage of the outcomes presented, such as the one developed by Capkun et al. [9], which provide a unique coordinate system by self-organization of the nodes without considering problems in location estimations. This kind of positioning systems can be naturally extended considering interference in RF signals caused by people's body.

References

1. Hightower, J. and Boriello, G. "Location systems for ubiquitous computing". *IEEE Computer* (August 2001) 57–66
2. Tseng, Y.C., Wu, S.L., Liao, W.H., and Chao, C.M. "Location awareness in ad hoc wireless mobile networks". *IEEE Computer*, Vol. 34, N. 6, (2001) 46–52
3. Bahl, P. and Padmanabhan, V. "RADAR: an in-building RF-based user location and tracking system", *Proceedings IEEE Infocom 2000*, IEEE CS Press, Los Alamitos, CA (2000) 775–784
4. Hills, A. "Wireless Andrew". *IEEE Spectrum*, Vol. 36, N. 6 (1999) 49–53
5. Want, R., Hopper, A., Falcão, V., and Gibbons, J. "The active badge location system". *ACM Trans. Information Systems* (1992) 91–102

6. Hightower, J., Want, R., and Boriello, G. "SpotON: an indoor 3d location sensing technology based on RF signal strength". UW CSE 2000-02-02, Univ. Washington, Seattle (2000)
7. Niculescu, D. and Nath, Badri. "Ad hoc positioning system (APS)". Rutgers University Technical Report DCS-TR-435. Rutgers University, New Jersey (2001)
8. AIR-PCM340 Client Adapter. In-Building: Cisco Aironet 340 Series Client Adapters. <http://www.baudia.fi/airpcm340.html>
http://www.cisco.com/en/US/products/hw/wireless/ps458/products_installation_guide_chapter09186a008007f74a.html
9. Capkun, S., Hamdi, M., and Hubaux, J.P. "GPS-free positioning in mobile ad-hoc networks". Cluster Computing Journal, Vol. 5, No.2 (2002) 157–167
10. Royer, E.M., and Toh, C.K. "A Review for Current Routing Protocols for Ad Hoc Mobile Wireless Networks", IEEE Personal Communications (April 1999) 46–55
11. Zou, Y., and Chakrabarty, K. "Sensor Deployment and Target Localization Based on Virtual Forces", Proc. IEEE INFOCOM (2003)
12. Elfes, A., "Occupancy Grids: A Stochastic Spatial Representation for Active Robot Perception", Proc. 6th Conference on Uncertainty in AI (1990) 60–70
13. Dube, R., Rais, C., Wang, K., and Tripathi, S. "Signal Stability based Adaptive Routing (SSA) for Ad-Hoc Mobile Networks," IEEE Personal Communication (1997) 36–45
14. Yusuke, M. Collaborative Environments for Disaster Relief. Master's thesis, Department of Civil and Environmental Engineering, MIT, Cambridge, MA (2001)

Mobility-Aware Rendezvous Point for Mobile Multicast Sources

Imed Romdhani¹, Mounir Kellil¹, Hong-Yon Lach¹, Abdelmadjid Bouabdallah²,
and Hatem Bettahar²

¹Centre de Recherche de Motorola. Paris, Espace Technologique St-Aubin,
91 193 Gif-sur-Yvette Cedex. France
{Imed.Romdhani, Mounir.Kellil, Hong-Yon.Lach}@Motorola.com

²HeuDiaSyc, CNRS UMR 6599, Université de Technologie de Compiègne
Centre de Recherches de Royallieu, BP 20529 - 60200 Compiègne Cedex. France.
{Bouabdal, Bettahar}@utc.fr

Abstract. The Internet research community has done a great effort to support mobile receivers in multicast session, but less interest was given to the problem of mobile sources. In fact, building source-specific tree with mobile sources is challenging. Depending on the nature of the mobility of the multicast source, several problems emerge such as the handover transparency, the multicast service interruption, and the reconstruction of the multicast delivery tree after each handover. In this work, we develop a new solution to make multicast source mobility transparent. Our solution introduces a new entity called Mobility-aware Rendezvous Point (MRP) in order to handle the mobility of a multicast source in both intra-domain and inter-domain multicasting. Our solution uses a smooth handover technique between MRP peers and interoperates efficiently with existing multicast routing protocols.

1 Introduction

To support IP mobility, the IETF (Internet Engineering Task Force) has proposed two protocols to handle the mobility of IPv4 hosts and IPv6 hosts: the Mobile IPv6 [10] and the Mobile IPv4 [6] protocols. These protocols aim to allow a mobile host to continue communicating with its correspondent hosts while moving. A mobile host can move from its home network to a foreign network without changing its *Home Address* (HoA). When a mobile host is away from its home network, it gets a new *Care-of Address* (CoA) from the visited network. The movement from one IP network to another is called a handover. To make this handover transparent to transport and higher-level communications, the mobile host maintains its HoA and registers the CoA with its *Home Agent* (HA). The association between the HoA and the CoA is called a *Binding*.

In the case of multiparty communications with IP multicast, the scenario of handover is particularly challenging. Multicast involves sending data to a restricted group of nodes and forms the basis for efficient implementation of multiparty applications on

the Internet. Studying and solving multicast issues in the stationary multicast infrastructure has been largely studied and many protocols have been proposed such as DVMRP, MOSPF, PIM and CBT [15]. However, fewer efforts have been focused on the specific problems of mobile receivers and sources. The basic difficulty of the IP multicast in the Mobile IP environment is the frequent change of membership and member's location [8][9][13].

When a mobile host is a source for a given multicast group, the mobility of the source is challenging. With the Mobile IP bi-directional tunnel approach (applicable for both the Mobile IPv4 and the Mobile IPv6 protocols), the mobile source can send its multicast data to the multicast group by using its home network infrastructure. Following this approach, when the mobile source is away from home, it uses its CoA as the source address to tunnel multicast packets to its Home Agent (HA). In return, the HA decapsulates the tunneled packets and forwards the enclosed data to the multicast delivery tree. The enclosed data contains the HoA as the source address and the multicast group address as the destination address. This approach is inefficient in packet delivery, waste system resources and result in long service latency, but it saves the transparency of the handover of the mobile source. With this approach, the entire multicast delivery tree is rooted on the source's home network and there is no need to reconstruct it whenever the handover occurs. With the remote approach, the mobile source can use the foreign network infrastructure to send multicast data to multicast group and thus avoid triangle routing across the home network. To do so, the mobile source uses its current CoA as the source address to send multicast packets to the local *Designated Router* (DR), which forwards them to the multicast delivery tree. Hence, the multicast delivery tree will be built with routing states that use the CoA and not the HoA as in the bi-directional approach. The main issue of the remote approach is that both multicast routers and receivers should be able to interpret the traffic coming from the CoA as coming from the same multicast source. In addition, the multicast routes should be updated to optimize multicast routing paths and avoid dropping multicast packets coming from a new source's CoA. Thus, with the remote approach, the handover of a multicast source is not transparent to multicast operations.

Some solutions have been proposed to handle mobile multicast sources [1][4][14]. In [1], author suggested a new RPF redirection mechanism to overcome the *Reverse Path Forwarding* (RPF) check failure in case of PIM-SM. In [4], authors introduced a new notification mechanism to notify multicast receivers to re-subscribe to the new *Source-Specific Multicast* (SSM) channel whenever the mobile multicast source changes its CoA. These solutions do not solve the problem of the transparency of the mobile source and do not avoid the reconstruction of the multicast delivery tree. The transparency problem occurs in the following scenarios:

- In case of shared-tree routing protocols such as PIM-SM [15], the *Rendezvous Point* (RP) of the multicast group will consider the source as a new sender whenever the source address changes.
- The source mobility is critical for protocols that construct source-specific trees (e.g. SSM [15]) because the reconstruction of the entire multicast delivery tree is expensive and inefficient for frequently mobile sources. For exam-

ple a protocol like DVMRP is clearly inefficient for a mobile source since the multicast tree must be recomputed after each source handover and packets are flooded to the entire network.

- If the mobile source always uses its current CoA, the border routers of inter-domain routing protocol such as DMSP peers (*Multicast Source Discovery Protocol*)[15] will interpret the mobile source as a new source. Furthermore, the mobility of the source increases the source activity announcement between border routers (i.e. by using *Source Active* (SA) messages).
- The receivers may encounter a multicast session disruption since they are not aware of the source's address change. If multicast receivers become aware of the CoA change and want to pursue the session, they have to register for this "new mobile source". Consequently a re-construction of the entire source-specific multicast tree is required.
- Since the IP multicast is receiver-initiated, when a mobile multicast source moves into a new foreign IP network, it cannot send immediately multicast data to the receivers until the source's Designated Router is notified of the existence of downstream receivers. This notification can be done for example by the MSNIP protocol (*Multicast Source Notification of Interest Protocol*)[15].

As the transparency is a crucial problem to support the multicast for mobile sources, we propose a new approach to avoid both the re-computation of the entire multicast tree and the notification of receivers about the handover. We introduce a new entity called **Mobility-Aware Rendezvous Point (MRP)**. The MRP is a core router, which is responsible for building a multicast mapping cache for mobile multicast sources. The MRP aims to make the handover of the source transparent to both on-tree routers and multicast receivers since the multicast routing states are built using the HoA and not the current CoA. Our solution works in intra-domain and inter-domain multicasting and can be used with existing multicast routing protocols such as PIM-SM and MSDP [2].

In this paper, we first present the problem statement and we investigate the limitations and the weakness of the current multicast routing protocols regarding the mobility of multicast sources in section 2. In section 3, we describe in details our MRP solution. The deployment scenarios and architectures of MRP in intra-domain and inter-domain are detailed in section 4. We conclude showing the advantages and the open issues of our MRP solution.

2 Problem Statement

In this section, we outline several problems of source mobility in IP multicast. A great effort was done by the Internet research community to support mobile receivers in multicast sessions [6][8][9][13], but less interest was given to the problem of mobile sources [1] [4][14]. We believe that the mobile source problems are more critical for a multicast session, since the handover of the source may affect the whole multicast

group communication, while the handover of a receiver has a local impact. To build source-specific tree, multicast routers have to keep source-specific multicast routing states. These states indicate the IP address of the source, the multicast group address, the upstream tree branch and eventually the downstream tree branches. Moreover, multicast routers exchange periodically keep-alive messages to capture state, topology, and membership changes. In a Mobile IP environment and depending on the nature of the source's mobility (micro-mobility or macro-mobility) several problems emerge:

- **Transparency:** the transparency is a major issue for mobile multicast sources [4][14]. When a mobile source moves from one IP network to another while it has an on-going multicast session, the mobile source gets a new IP address (CoA) from the new visited network. Then, the mobile source uses the new CoA as a source address to route its multicast data. Unfortunately, the new source's DR cannot forward the multicast packets sent with the new CoA address until the receivers explicitly notify it. Without this notification, multicast data will not be forwarded downstream. In addition, multicast receivers are not able to interpret the traffic coming from the new CoA as coming from the same source.
- **Reverse Path Forwarding (RPF):** the RPF check compares the packet source's address against the interface upon which the packet is received. When the handover occurs, the mobile multicast source can not use its HoA as the source address to route multicast packets from the foreign network due to the ingress filtering problem and the RPF check failure.
- **Packet loss:** during the migration from one source-specific tree to another due to the source's IP address change, the mobile multicast source might not send multicast packets unceasingly while moving. Due to tree migration, multicast receiver may miss some multicast packets.
- **Multicast scoping:** the scoping is a technique that can be used to limit and control the propagation of the multicast traffic by configuring it to an administratively defined topological area (domain, site, local subnet, etc.). The multicast scoping is required to relieve stress on scarce resources, such as bandwidth, and to improve privacy or scaling properties. However, when coupled with source mobility, the mobile source can be "out of the scope" of the multicast group since it will be located out of the range of the multicast traffic distribution. The source's border router in the previous network (i.e. prior to the handover) can filter incoming multicast packets and denies access to its local groups since the mobile source is considered as a foreign source. Hence, receivers will not receive multicast data anymore.
- **Inter-domain problem:** the inter-domain multicast branches have to be re-constructed whenever the source moves into a new domain. An extra signaling mechanism is also required by border multicast routers to exchange the new source's address and to dynamically update their inter-domain multicast forwarding tables accordingly. Border multicast routers have to know permanently which are the active sources for all the served groups. For example, in the case of the MSDP protocol, the mobility of the multicast source within a PIM-SM domain will cause a "source active inundation problem". In fact the

MSDP peers have to increase the source announcements (by using Source Active message) to reflect the new source's IP address. As shown in Figure 1, MSDP peer 1 has to send one SA message per handover to all other MSDP peers (i.e. SA(CoA2,G,MSDP1), SA(CoA2,G, MSDP1), etc.).

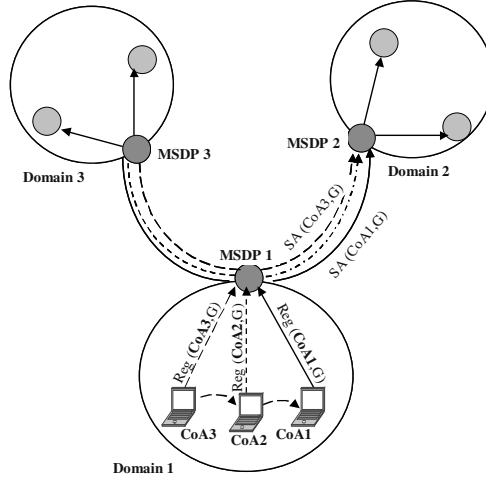


Fig. 1. Multicast source micro-mobility impacts

The Internet research community has done a great effort to support mobile receivers in multicast session [6][8][9][13], but less interest was given to the problem of mobile sources [1][4][14]. We believe that the mobile source problems are more critical for a multicast session, since the handover of the source may affect the whole multicast group communication, while the handover of a receiver has a local impact. In the literature there are three solutions that have been suggested to overcome the problems caused by the mobility of a multicast source. The first solution uses an explicit notification to notify multicast receivers about source handover [4]. The receivers have to trigger a new join to start building a new specific-tree using the new CoA. The second solution attempts to overcome the problem of the RPF check when using the PIM-SM protocol [1]. The third solution suggests a framework that supports multicast over Mobile IP. This framework is designed for the case when the mobile host is a sender as well as a receiver for a multicast group [14].

In this work, we develop a new solution in order to make multicast source mobility transparent. Our solution introduces a new entity called **Mobility-aware Rendezvous Point (MRP)**. Our primary goal is to handle efficiently the mobility of a multicast source in both intra-domain and inter-domain multicasting. We prove that our solution interoperates with existing multicast routing protocols such as PIM-SM and MSDP.

3 Our Solution: Mobility-Aware Rendezvous Point

3.1 General Idea of Our Solution

In order to support transparent multicast source mobility and avoid the re-computation of the multicast tree, we introduce a new entity called Mobility-Aware Rendezvous Point (MRP). The MRP is the meeting point where multicast receivers meet mobile source. The MRP builds a **Multicast Registration Cache (MRC)** for mobile multicast sources. This cache is used to map the permanent *Home Address* (HoA) of a mobile source with its temporary *Care-of Address* (CoA). Based on the MRC information, we propose a new **Multicast Forwarding Table (MFT)** format in which each multicast source will be referenced by the two addresses (HoA and CoA) instead of a single IP address. This solution introduces a new registration method for IP mobile multicast source. Compared to basic PIM-SM registering procedure, in our approach, the mobile source registers only once with the MRP by sending a Source Registration (SR).

To send multicast data, the mobile multicast source encapsulates its data packets and sends them to the MRP. To do so, the mobile source uses two IP headers. The first header contains the current CoA as the source address and the multicast group address G as the destination address, whereas the second header contains the HoA and the group address (G). Before forwarding these encapsulated packets, the source's MRP checks first if these packets are coming from a registered and trusted mobile multicast source or not. If so, the source's MRP decapsulates these packets and sends the enclosed data with the (HoA, G) header to the multicast receivers. When the mobile source moves to a new IP subnet within the MRP service area, the source's MRP is implicitly notified about the CoA change.

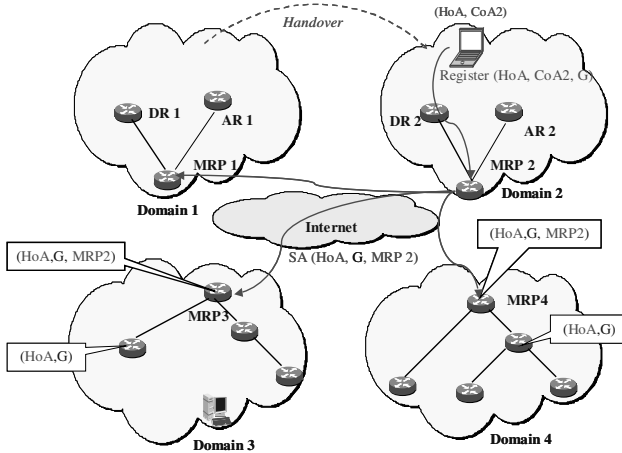


Fig. 2. MRP architecture

In case of inter-domain multicasting, if the source moves to a new domain, it has to register again with the local MRP in the new domain. The new MRP notifies remote MRPs about the source address change. The notification can be done for example by the source announcement mechanism of MSDP. As shown in Figure 2, the source-specific trees built in foreign networks 3 and 4 are constructed with (HoA, G) states and not with (CoA1, G) states. Since MRP3 and MRP4 are unaware of the mobility of the source, they will not reconstruct the source-specific tree within their domains. Hence, only the branches (typically IP tunnels) between MRPs (MRP2, MRP3, MRP4) are reconstructed and not the entire delivery tree.

To avoid the service disruption during the handover, the mobile source has to quickly discover the new MRP (NewMRP) in the new visited IP network. Then it notifies its previous MRP (OldMRP) about the CoA change. Depending on the source's movement detection approach and the mechanism used for the acquisition of a new CoA, the source can continue to use its OldMRP while registering with the NewMRP. In this case, remote MRPs peers will still receive multicast data from the OldMRP. Upon reception of the first multicast packet from the new visited network, the MRPs peers will switch to the NewMRP.

Thanks to the mobility awareness, intra-domain multicast tree rooted on the MRP will not be reconstructed after the handover of the mobile source. In addition, the notification of multicast receivers is therefore not required and the transparency of the handover is guaranteed.

Table 1. Data structure of the SR message

Fields	Description
Source Address	CoA
Destination Address	The local MRP
Mobility Flag	A mobility flag (M=1) is explicitly set to 1 to indicate that the source is a mobile capable node. This flag is set to 0 if the source is a stationary host.
Router Flag	This flag is optional and it is valid if the multicast source is a router capable node (reserved for future use: case of Mobile Networks).
The Previous MRP Address	The previous MRP address is given (if the source is already registered with another MRP. This field is optional and it is useful for seamless handover purpose).
Registration Life Time	This lifetime may be proportional to the multicast session lifetime announced by the mobile source, the CoA lifetime, and the multicast address lifetime.
Sequence Number	The sequence number of the last registration request sent to the MRP.
Recent Usage	Update or creation
Security Association	The Registration Security Association (RSA) is optional and it is used for security purpose.
Multicast Address	The set of IP multicast group addresses if the mobile source is a sender for different IP multicast groups.
The Multicast Group Security Key(s)	Optional

3.2 Detailed Description

The MRP will be the root point of a single delivery tree for a given IP multicast group and a mobile source. In our approach, the mobile source registers with the local MRP by sending a **Source Registration (SR)** message. The SR message can be forwarded by the local Designed Router (DR) or sent directly by the mobile source to the closet MRP. With this message, the mobile source requests to send multicast data to a given multicast group for a fixed period of time that may be a trade-off between the multicast address lifetime, the CoA lifetime, and the session lifetime. Table 1 summarizes the data structure of the SR message.

Upon reception of the SR message, the MRP may accept or reject the source's request. If the MRP accepts the registration, it caches this registration in its Multicast Registration Cache (MRC) and replies with a **Registration Acknowledgment (Reg_Ack)** message to the mobile source. The Reg_Ack message contains the MRP's address, the *Registration Security Association (RSA)* and the remaining registration lifetime. Then, the MRP updates its *Multicast Forwarding Table (MFT)*. When a MRP first learns of a new mobile source it constructs a Source Active message that will include the source's HoA. Then sends this SA message to its MRP peers in other domains. If a remote MRP peer has group members within its domain, it triggers a (HoA, G, Source's MRP) join message towards the source's MRP. Thanks to the new SA message, the MRP peers will not initiate the reconstruction of their own intra-domain source-specific trees.

In the next sections, we describe how our solution supports both intra-domain and inter-domain multicasting.

3.3 Intra-domain and Inter-domain Multicasting with MRP

The MRP entity helps to make the movement of the mobile source transparent to the multicast receivers. By using the two addresses (HoA and CoA), the notification of multicast receivers is therefore not required. In this section, we discuss several architecture scenarios about how to deploy MRP entities. For each architecture, we outline its strengths and limits.

3.3.1 Architecture for Intra-domain Multicasting

The simple architecture is to use a single MRP router. Before sending data to a multicast group, the multicast source has to register with the MRP by sending a source registration message. This message instantiates the construction of a source-specific branch with (Current CoA, G) states between the source's DR and the MRP. In return, the MRP forwards multicast data downstream the multicast tree delivery tree. The MRP rooted tree is built with (HoA, G) states, whereas the specific branch is built with (CoA, G) states. Whenever the handover occurs, only the source specific branch has to be reconstructed. This intra-domain architecture is simple and scalable. With redundant MRPs, the single point of failure and bottleneck problem can be avoided. How-

ever, the discovery of the optimal MRP location is the major open issue [3][11]. Table 2 summarizes the similarities and the differences between the MRP architecture and the PIM-SM one.

Table 2. Comparison between PIM-SM and MRP

	PIM-SM	MRP
Source Registration Process	Construction of a source-specific tree between the source's DR and the RP	Construction of a source-specific branch (between the source's DR and the MRP).
Join Process	Receivers have to rejoin the multicast delivery tree whenever the handover occurs. As a consequence, the multicast routers have to send new (CoA, G) join messages.	Receivers have to join once the multicast delivery tree. Multicast router use (HoA,G) join messages instead of (CoA, G) join messages.
Tree Reconstruction	The RP rooted tree is reconstructed whenever the handover occurs.	No reconstruction: the MRP rooted tree is not vulnerable to the handover of the mobile source.

3.3.2 Architecture for Inter-domain Multicasting

To support inter-domain multicasting with mobile sources, our solution handles efficiently the macro-mobility and reduces significantly the signaling process between IP domains. To do so, our architecture relies on several MRPs. Each MRP is within a single IP domain. To support source mobility, a signaling mechanism is required between MRP peers. The MRP peers have to exchange source active and keep-alive messages to track the source's MRP attachment point changes. Figure 3 and Figure 4 illustrate how this architecture works. In this example, (HoA, CoA1) pair identifies respectively the IP address of the mobile multicast source in its home and foreign networks. The mobile multicast source first sends a Source Registration message $\text{Reg}(\text{HoA}, \text{CoA1}, \text{G})$ to the MRP1 in domain 1. Initially, the MRP1's MFT is empty. When the MRP1 acknowledges the registration, it replies with a $\text{Reg_Ack}(\text{MRP1})$ message. Then, it updates its MFT by adding a new entry for the mobile source. After that, the MRP1 sends three $\text{SA}(\text{HoA}, \text{CoA1}, \text{MRP1})$ messages to MRP2, MRP3 and MRP4 located respectively in domain 2, domain 3 and domain 4. In our example, MRP2 has no interested receiver, but MRP3 and MRP4 have multicast members that wish to join the multicast session identified by (HoA, G). The MRP3 and MRP4 have their own (HoA, G) source-specific trees and send Join (HoA,G) message towards MRP1 to construct a source-specific branch with it.

To send multicast data, the mobile multicast source encapsulates its data packets and sends them to the MRP1. To do so, the mobile source uses two IP headers. The outside header is set to (CoA1, G), where CoA1 is the current source's address and G is the multicast group address. The inside header is set to (HoA, G). Before forwarding these encapsulated packets, the MRP1 checks first if these packets are coming from a registered and trusted mobile multicast source or not. If so, it decapsulates

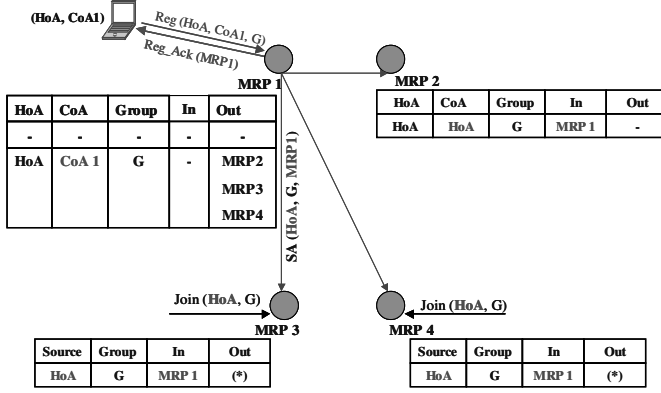


Fig. 3. Multicast forwarding table of MRP peers

them and sends the enclosed data with the (HoA, G) header to both MRP3 and MRP4. While the source is roaming within the MRP1 domain, MRP1 does not flood new Source Active messages towards remote MRP peers whenever the CoA changes. Only one SA message is required. Compared to the MSDP protocol, when the source changes its IP address, the source's RP has to construct a new Source Active message per handover, which sends it to MSDP peers. Hence, our solution is more effective since it hides the intra-domain mobility of mobile sources and reduces significantly the signalling process between MRP peers.

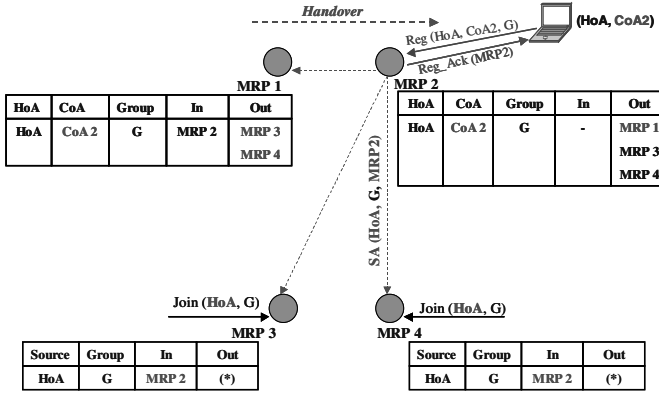


Fig. 4. MFTs after the handover of the mobile source

When the mobile source moves to the MRP2's network, it registers again with MRP2. The MRP2 updates its MFT and sends (HoA, G, MRP2) SA message to MRP1, MRP3 and MRP4, which update their MFTs. During the construction of new branches to reflect the new source's MRP location, the mobile source may continue to send its multicast data through MRP1. When MRP3 and MRP4 start receiving multicast data from MRP2, they invalidate the old branches with MRP1. The two entrees that reference the MRP1 and MRP2 in the MFT of MRP1 coexist for a short period of time in

order to assume a seamless handover of the mobile source. By this way, multicast service disruption can be avoided during this period.

In our architecture, source-specific tree between RP peers may be constructed instead of using IP tunnels. To construct such tree, both (HoA, G, source's MRP) join messages and peer-RPF check mechanism are required. Unlike the RPF check, which compares the packet source's address against the interface upon which the packet is received, the peer-RPF check compares the MRP address carried in the SA message against the MRP peer from which the message was received. If a remote MRP peer has group members within its domain, it triggers a (HoA, G, Source's MRP) message towards the source's MRP. Each MRP peer has to keep awareness about source's MRP change and not source's IP address change.

3.4 Discussions

Our new solution has several key features. First, it offers an optimal routing for mobile multicast sources and makes their handover transparent to the multicast receivers. Second, handover of mobile sources is seamless and accompanied with minimal packets loss. Third, the mobile source registers once with the local MRP and the registration procedure is simple. Moreover, our new solution interoperates very well with both existing intra-domain and inter-domain multicast routing protocols regardless whether the multicast sources are stationary or mobile nodes. The other major benefit of our architecture is that it is applicable to both the Mobile IPv4 and the Mobile IPv6 environments. Existing multicast protocols such as PIM-SM and CBT can be easily extended to be used with our MRP architecture. We only need fewer modifications of the multicast signaling protocol and the core routers functionalities to implement our solution.

4 Conclusion

Supporting multicast with mobile nodes is a serious issue to warrant enough attention. The challenges are inherited from the IP mobility handover impacts and the source-specific tree requirements. In this paper we, focused on the issue of the source mobility transparency in order to make the handover of IP mobile sources transparent to both the main multicast delivery tree and to the multicast receivers. To do so, our solution combines the simplicity feature of the optimal routing of shared tree with minimal signaling mechanism. We proposed a new Mobility-Aware Rendezvous Point (MRP) and we discussed the different architectures to deploy the MRP entities for both intra-domain and inter-domain multicasting. We are looking forward to simulate and implement our solution using the Livsix IPv6 stack [16].

References

1. A O'Neill, "Mobility Management and IP Multicast", Internet Draft, draft-oneill-mip-multicast-01.txt, 2002.
2. Bill Fenner, and David Meyer, "Multicast Source Discovery Protocol (MSDP)", IETF Internet Draft, draft-ietf-msdp-spec-20.txt, May 2003.
3. C. Donahoo and Zegura, "Core Selection methods for multicast routing", in Proc. ICCCN, 1996.
4. C. Jelger and T. Noel, "Supporting Mobile SSM Sources", Globecom'02, IEEE Global Communications Conference, Taipei, Taiwan, November 2002.
5. C. Jelger and T. Noel, "Multicast for Mobile Hosts in IP Networks: Progress and Challenges", IEEE Wireless Communications, October 2002.
6. C. Perkins, "IPv4 Mobility Support", RFC 2002, October 1996.
7. Christian Bettsteller, Anton Riedl, and Gerhard Ge(ler, "Interoperation of Mobile IPv6 and Protocol Independent Multicast Dense Mode", ICPP Workshop 2000, pp. 531–540, 2000.
8. Chunhung Richard Lin and Kai-Min Wang, "Mobile Multicast Support in IP Networks", IEEE INFOCOM 2000, pp. 1664–1672, March 2000.
9. Chunhung Richard Lin, "Scalable Multicast Protocol in IP-based Mobile Networks", Wireless Networks, 8, 27–36, 2002.
10. David B. Johnson, C. Perkins, "Mobility support in IPv6", IETF Internet Draft, draft-ietf-mobileip-ipv6-24.txt, June, 2003.
11. Eric Fleuy, Yih Huang, and Philippe K. McKinley, "On the Performance and Feasibility of Multicast Core Selection Heuristics", Networks, John Wiley, 2000.
12. George Xylomenos and George C. Polyzos, "IP Multicast for Mobile Hosts", IEEE Communications Magazine, Volume 35, Number 1, pp. 54–58, 1997.
13. Hrishikesh Gossain, Carlos de Morais Cordeiro, and Dharma P. Agrawal, "Multicast: Wired to Wireless", IEEE Communications Magazine, June 2002.
14. Hrishikesh Gossain, Siddesh Kamat, and Dharma P. Agrawal, "A Framework for Handling Multicast Source Movement over Mobile IP", ICC 02, May 2, 2002.
15. Maria Ramalho, "Intra- and Inter-Domain Multicast Routing Protocols: A survey and taxonomy", IEEE Communications Surveys & Tutorials, First Quarter 2000, vol.3 no.1.
16. Networking and Applications Lab (NAL), "Livsix Release 2", Centre de Recherche de Motorola - Paris (CRM), Motorola Labs, December 2002.
17. Vineet Chikarmane, Carey L. Williamson, Richard B. Bunt and Wayne L. Mackrell, "Multicast support for mobile hosts using mobile IP: design issues and proposed architecture (MoM protocol)", ACM/Baltzer Mobile Network and Applications, vol. 3, no. 4, pp. 365–379, 1999.
18. Yu Wang and Weidong Chen, "Supporting IP Multicast for Mobile Hosts", Southern Methodist University, May 8, 1998.

An Address Configuration and Confirmation Scheme for Seamless Mobility Support in IPv6 Network^{*}

Seung-Hee Hwang¹, Youn-Hee Han², Sung-Gi Min¹, and Chong-Sun Hwang¹

¹ Department of Computer Science and Engineering, Korea University, Seoul, Korea
{shhwang,hwang}@disys.korea.ac.kr, sgmin@korea.ac.kr,

² i-Networking Lab., Samsung AIT, Yongin, Kyungki-do, Korea
yh1.han@samsung.com

Abstract. IETF Mobile IP version 6 and its fast handover protocol are proposed to handle routing of packets to a mobile node when it has moved away from its home network. To do this, each time a mobile node moves to a new location, it configures and confirms its temporal IP address. In this paper, we study the impact of the address configuration and confirmation procedure on the IP handover latency. We first argue that the current strategies for the address configuration and confirmation are unnecessarily slow, so they prevent current protocols (e.g. MIPv6) from being used for real-time traffic. We present a new scheme that can completely eliminate the latency taken by the address configuration and confirmation from the handover latency, so it can be a substitute of the current strategies. A mathematical analysis is developed to show the benefits of our scheme. In the analysis, we compare our scheme with the existing proposals in terms of the handover latency.

1 Introduction

Mobile IPv6 (MIPv6) [1,2] has been proposed in the IETF to accommodate the increasing demand of mobility in the Internet. A Mobile node (MN) can move freely anywhere, since it informs its mobility manager called home agent (HA) of its new location represented by a temporal IP address in the protocol. The HA manages a pair of the MN's original IP address as known home address (HoA) and its temporal IP address called Care-of Address (CoA). It can inform correspondent nodes (CNs) which want to communicate with the MN of the CoA. Especially CNs on communication with the MN can be notified its CoA by the MN directly, and the communication can be on going. Because the uniqueness of CoA or the IP address of MN plays very important role in MIPv6, that should be assured before communication.

According to the proposal, an MN should generate a new care-of address (CoA) using IPv6 stateless (or stateful) address auto-configuration whenever it moves to a new link. To verify the uniqueness of this CoA, the MN runs *duplicate*

^{*} This research was supported by University IT Research Center Project.

address detection (DAD) algorithm [3] (hereinafter, it is called RFC 2462 DAD) before assigning the address to its interface. The algorithm checks if the address chosen by an MN is already in use using neighbor discovery procedures. That is, once the CoA has been configured, the MN sends a solicitation message toward that address and waits the response of the message. If another node has been configured with the same address, it will reply and advertise its address, exposing the collision. At this point, the MN will know that its address is not unique. After that, it will configure another address and retry the DAD procedure, or it will display an error message. If the MN does not receive any reply for a period (at least 1000ms according to [1.]), it considers its address unique, and it can use this address. An MN must perform this DAD procedure every time it handovers between IPv6 networks, and it cannot begin communication until the procedure completes. After successful DAD, it registers the new CoA (NCoA) to its HA and CNs using binding update (BU) messages.

Recently several performance issues have been identified with handover latency in MIPv6, notably the duration which is taken to automatically configure NCoA and confirm its uniqueness. The current DAD algorithm [3] takes at least 1000ms to be completed. Obviously, the DAD is a time consuming process, particularly when an MN in need of seamless handover runs it. Optimistic DAD (oDAD) [9] has been proposed for eliminating the DAD completion time. oDAD is based on the premise that a DAD is far more likely to succeed than fail; nevertheless, there is still the probability of address collision and it should be avoided.

FMIPv6 (Fast Handover for Mobile IPv6) [4] is proposed to reduce the handover latency in MIPv6 [1]. It describes a protocol to replace MIPv6 movement detection algorithm and NCoA configuration procedure for fast handover. It enables MN to quickly detect that it is now moving to a new subnet by providing the new access point (AP) identifier and receiving the associated subnet prefix information. The MN formulates a prospective NCoA on current subnet from the information. To allocate the NCoA to the MN's interface immediately after attaching to the new subnet, FMIPv6 provides the NCoA confirmation procedure, which is executed before or during MN switches its subnet.

The FMIPv6 consists of two handover modes: 'predictive' and 'reactive' mode. The scenario in which an MN receives the positive result about the confirmation of its prospective NCoA on the current subnet is called 'predictive' mode. The scenario in which an MN checks the uniqueness of NCoA after the MN attaches to a new subnet is called 'reactive' mode. Although the MN early initiates the NCoA confirmation on the current subnet, FMIPv6 would fall into the reactive mode if the MN could not receive the confirmation result on the current subnet. In addition, if the proposed NCoA is rejected during the NCoA confirmation procedure, the MN may configure another NCoA by itself so that handover latency becomes long. In order to achieve more reduction of handover latency, it is required that 'predictive' mode should occur more frequently than 'reactive' mode. So, the NCoA confirmation should be done promptly and the result should be always positive.

In this paper, we propose a new address configuration and confirmation scheme that completely takes off DAD procedure/time from L3 handover procedure/latency. This is achieved by an advance configuration of NCoA, which will be used without any concern about address collision after an MN moves to a new link. With a little modification of handover procedures for the two representative protocols: MIPv6 and FMIPv6, our scheme can make them improved in terms of handover latency.

This paper is organized as follows: in Section 2, we will provide our new scheme and present the procedures for our scheme's application to the existing two protocols: MIPv6 and FMIPv6; in Section 3, we will show a performance improvement using a mathematical analysis in case that the proposed scheme is applied into the protocols; and we will conclude our work with several words in Section 4.

2 New Address Configuration and Confirmation Scheme

The goal of our proposal is to allow an MN to obtain a duplication-free NCoA before or during it establishes connectivity with a new access router (NAR). In this section, we will describe our proposal, named '*Advance DAD (aDAD)*', in detail. Our proposal consists of the following three steps

1) [Generation of random addresses] If an AR has an interface connected to a subnet where an MN can move, it must manage a 'Passive Proxy Cache' associated with the interface. The AR randomly generates globally routable addresses as background process. The number of addresses generated initially is at least *CapacityProxyCache*. By default, the *CapacityProxyCache* is 100, but it should be configured based on the router capacity and the expected MNs' solicitation load. The default random address generation procedure is similar with one described in RFC 3041 [5]; however, another procedure, if any, can be used for the address generation.

2) [Check of uniqueness for the generated addresses] On generating an address, an AR must perform RFC 2462 DAD on the address. Only after successful DAD, the AR reserves the address into its 'Passive Proxy Cache' and regards the reserved address as a duplication-free NCoA. The AR should try to generate a new address so that the number of addresses reserved in 'Passive Proxy Cache' becomes *CapacityProxyCache*.

3) [Guarantee of their perpetual uniqueness] As soon as a duplication-free NCoA is stored, an AR begins to act as a proxy for the address '*passively*'. While the AR is serving as a Passive Proxy for an address, it must not multicast onto its link an unsolicited Neighbor Advertisement (NA) message [3] with the address. This behavior's reason is because the AR does not have to intercept packets delivered to the address; moreover, such packets may not be created or delivered from any node until an MN uses it as its NCoA. In addition, the AR must not reply to a Neighbor Solicitation (NS) message with a Target Address field set to one of addresses stored at the 'Passive Proxy Cache'. When the AR receives such an NS with the purpose of DAD executed from any other

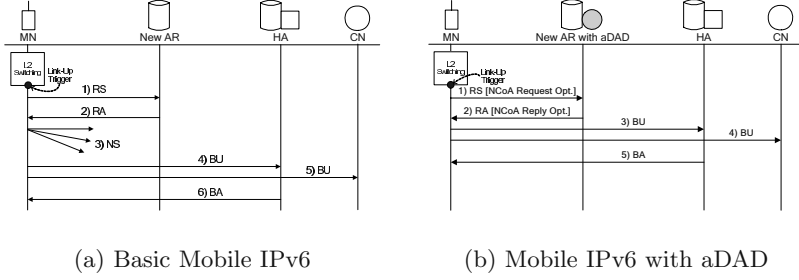


Fig. 1. Application of Advance DAD to Mobile IPv6

node¹, it must silently delete the address specified in the Target Address field from its 'Passive Proxy Cache'. At any time of such deletion, the AR should generate a new address and keep the total number of addresses in the cache being *CapacityProxyCache*. By obeying these rules, AR can keep the reserved addresses unique lastingly until they are allocated into an MN.

2.1 Application of Advance DAD to MIPv6

MIPv6 supports the mobility of IPv6 nodes by using two IP addresses: a home address (HoA) which represents the fixed address of the node and a CoA which changes with the subnet that the node is currently attached to. In MIPv6, each CN can have its own 'binding cache' where HoA plus CoA pairs are stored. So, a CN is able to send packets directly to an MN, when the CN has a recent entry for the MN in its binding cache.

We assume that an MN's Layer 3 stack receives 'Link-Up trigger' from the Layer 2 stack when it arrives on a new L2 link. The Link-Up trigger allows the MN to quickly initiate signaling with an NAR. On moving to a new AR's subnet, with help of the Link-Up trigger, the MN sends a Router Solicitation (RS) message to the All-Routers multicast address. An AR receiving the RS will send a Router Advertisement (RA) message to the MN. In the basic MIPv6 (Fig. 1 (a)), the MN itself configures an NCoA by combining the network prefix carried by the RA and its interface identifier. And then, it must run RFC 2462 DAD, thereby it sends an NS message to the Solicited-Nodes multicast address of its current tentative address and waits for 1000 ms.

In MIPv6 supported with 'aDAD' (Fig. 1 (b)), an MN sends an RS with an 'NCoA Request option'. This new option notifies the AR of the request of a duplication-free NCoA, and includes the MN's Previous CoA (PCoA) and a link-layer address. If the AR acting as a Passive Proxy receives the RS with the 'NCoA Request option', it follows the following procedures:

- 1) select an address from the 'Passive Proxy Cache' and delete it from the cache.

¹ in such case, the Source Address field in the NS message is an unspecified address.

- 2) create a neighbor cache entry using the selected NCoA and the MN's link-layer address.
- 3) create a host route entry using the MN's PCoA and the link-layer address.
- 4) create an RA with an 'NCoA Reply option'. The NCoA Reply option includes the address selected from the 'Passive Proxy Cache'.
- 5) using the host route entry, send the RA with the 'NCoA Reply option' directly to the MN's PCoA.
- 6) immediately delete the host route entry.

As soon as an MN receives an RA with an 'NCoA Reply option', it configures the address specified in the option as its NCoA. It should be noted that the MN does not have to run RFC 2462 DAD on the address. Also, using the address, the MN can immediately send a BU message to the HA and CNs.

2.2 Application of Advance DAD to FMIPv6

In FMIPv6, fast-handover initiation is based on a Layer 2 trigger (or a Layer 2 signal), which informs that an MN will soon be handed over. To start a fast-handover, the MN sends a Router Solicitation for Proxy (RtSolPr) message to the previous access router (PAR). The RtSolPr contains a Layer 2 identifier of a target AP which the MN moves to. At this time, the PAR maps the Layer 2 identifier into a proper target NAR. The MN will receive a Proxy Router Advertisement (PrRtAdv) message in response from the PAR with the NAR's network prefix. Based on the response, the MN forms a prospective NCoA and immediately sends the NAR a Fast Binding Update (FBU) message with the prospective NCoA. When the PAR receives an FBU, it immediately sends a Handover Initiation (HI) message to the NAR. This message initiates the process of establishing a bidirectional tunnel between PAR and NAR (see Fig. 2).

This HI contains the prospective NCoA, and the NAR starts its NCoA confirmation procedure as soon as it receives the HI message. If the period of confirmation procedure would be long, (hereby, the delivery of a Handover Acknowledgement (HACK) and a Fast Binding Acknowledgement (FBACK) would be delayed), the MN may not receive the FBACK before it disconnects with the PAR (see Fig. 2 (a)). This means that FMIPv6 falls into the reactive mode, so the MN should (re)send an FBU as soon as it attaches to the NAR. As a result, it requires the additional FBU delivery, which will be encapsulated in a Fast Neighbor Advertisement (FNA), with the consumption of wireless bandwidth.

If the NAR receives the FNA and an encapsulated FBU and detects that NCoA is already used by other nodes, it must discard the inner FBU and notify this fact of the MN (see Fig. 2 (b)). Obviously, it causes handover latency to be extended. This case can be occurred even when the period of confirmation procedure is very short (thereby, an FBACK is delivered to the MN prior to disconnecting with the PAR (see Fig. 2 (b)^{2,3})). If the result of confirmation shows

² In Fig. 2 (b), the second FBU is delivered from MN to PAR via NAR

³ In Fig. 2 (b), the packets are directly tunneled to MN's NCoA

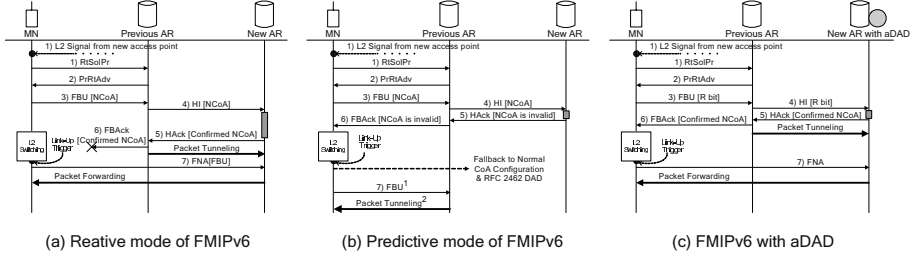


Fig. 2. Application of 'Advance DAD' to FMIPv6

that the prospective NCoA is invalid (not unique), MN should itself configure its NCoA and run RFC 2462 DAD after moving to NAR and before sending FBU.⁴

If the NAR adopts 'aDAD', then these problems can be clearly eliminated (see Fig. 2 (c)). In our proposal, an MN does not need to configure its prospective NCoA when it receives a PrRtAdv. Instead of configuring an NCoA, the MN sends the PAR an FBU with a newly defined flag, named 'NCoA Request bit (R bit)'. If an NAR acting as a Passive Proxy receives an HI with the 'R bit' set, it follows the following procedures:

- 1) select a duplication-free NCoA from the 'Passive Proxy Cache' and delete one from the cache.
- 2) create a neighbor cache entry using the selected NCoA and the MN's link-layer address retrieved from the HI message's option [4].
- 3) create an HAck with a 'NCoA Reply option', and send it to the PAR. The 'NCoA Reply option' includes the selected duplication-free NCoA.

The PAR sends the MN an FBack with the 'NCoA Reply option' contained in the HAck. As soon as the MN receives the FBack, it configures the address specified in the 'NCoA Reply option' as its NCoA. In the proposal, it takes very short time to configure and confirm the NCoA. Therefore, FMIPv6 will be operated as the preactive mode in most cases.

3 Performance Analysis

We use a simple model for the data packet traffic. During a session, several packets are generated by a CN at a constant rate and they reach an MN at the same rate. We assume that the session duration time t_o has the exponential distribution with mean $E[t_o] = 1/\lambda_o$.

We will calculate the total handover latency per session for each protocol: MIPv6 and FMIPv6 with/without a DAD. The handover latency is defined for a receiving MN as the time that elapses between the disconnection with the previous attachment of point and the arrival of the first packet after the MN moves to NAR.

⁴ FBack may provide an alternative and confirmed NCoA. However, the current specification of FMIPv6 presents no method about this.

3.1 System and Mobility Model

We assume the homogeneous network of which all wireless access point (AP) areas in a subnet have the same shape and size. Let t_p and t_s be i.i.d. random variables representing the AP area residence time and the subnet residence time, respectively. Let $f_p(t)$ and $f_s(t)$ be the density function of t_p and t_s , respectively. Suppose that an MN visits k AP areas in a subnet for a period t_s^k . During t_s^k , the MN resides at AP area i for a period t_i . Then, $t_s^k = t_1 + t_2 + \dots + t_k$ has the following density function.

$$f_s^{(k)}(t) = \int_{t_1=0}^t \int_{t_2=0}^{t-t_1} \dots \int_{t_{k-1}=0}^{t-t_1-\dots-t_{k-2}} f_p(t_1) f_p(t_2) \dots f_p(t_{k-1}) f_p(t-t_1-t_2-\dots-t_{k-1}) dt_{k-1} dt_2 dt_1. \quad (1)$$

Using the Laplace transform convolution, we can get the Laplace transform for $f_s^{(k)}(t)$ as follows.

$$f_s^{(k)*}(a) = [f_p^*(a)]^k, \quad (2)$$

where $f_p^*(a)$ is the Laplace transform of $f_p(t)$.

We describe a two-dimensional random walk model for mesh planes in order to compute the subnet residence time density function. Our model is similar to [6], but it considers a regular AP area/subnet overlay structure. We assume that an MN resides in an AP area for a period and moves to one of its four neighbors with the same probability, i.e., with probability $1/4$. A subnet is referred to as an n -layer subnet if it overlays with $N = 4n^2 + 4n + 1$ AP areas. Fig. 3 (a) shows the 3-layer subnet. The AP areas that surround layer $x-1$ AP areas are called *layer x AP areas*. An n -layer subnet overlays AP areas from layer 0 to layer $n-1$. The AP areas that surround the layer $n-1$ AP areas are referred to as *boundary neighbors*, which are outside of the subnet.

According to the equal moving probability assumption, we classify the AP areas in a subnet into several AP area types. An AP area type is of the form $\langle x, y \rangle$, where x indicates that the AP area is in layer x and y represents the $y+1$ st type in layer x . AP areas of the same type have the same movement flow pattern. Fig. 3 (a) also illustrates the type of AP areas for 3-layer subnet.

In the random walk model, a state (x, y) represents that the MN is in one of the AP areas of type $\langle x, y \rangle$. The absorbing state (n, j) represents that an MN moves out of the subnet from state $(n-1, j)$, where $0 \leq j \leq 2n-3$. The state diagram of the random walk for 3-layer subnet is shown in Fig. 3 (b). The *transition matrix* of this random walk is $P = (p_{(x,y)(x',y')})$. We use the Chapman-Kolmogorov equation to compute $P^{(k)}$. For $k \geq 1$, an element $p_{(x,y)(x',y')}^{(k)}$ in $P^{(k)}$ is the probability that the random walk moves from state (x, y) to state (x', y') with exact k steps. We also define $p_{k,(x,y)(n,j)}$ as

$$p_{k,(x,y)(n,j)} = \begin{cases} p_{(x,y)(n,j)} & \text{for } k = 1 \\ p_{(x,y)(n,j)}^{(k)} - p_{(x,y)(n,j)}^{(k-1)} & \text{for } k > 1. \end{cases} \quad (3)$$

Then, $p_{k,(x,y)(n,j)}$ is the probability that an MN initially resides at a $\langle x, y \rangle$ AP area, moves into a $\langle n-1, j \rangle$ AP area at the exact $k-1$ st step, and then moves out of the subnet at the k th step.

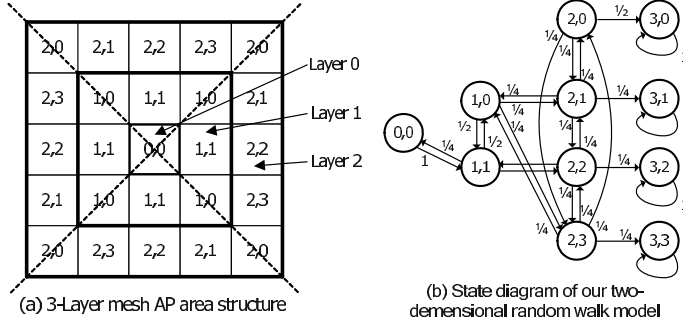


Fig. 3. Mesh AP area structure and the state diagram of our 2D random walk model

Let $q_{(n-1,j)}$ be the probability that an MN enters the subnet through a $\langle n-1, j \rangle$ AP area at the first step. Then, for $n \geq 2$, From [7,10], for 3-layer subnet, we can have $q(2,0) \simeq 40\%$ and $q(2,j) \simeq 20\%$ for $0 < j \leq 3$. For 4-layer subnet, we have $q(3,0) \simeq 28.571\%$ and $q(3,j) \simeq 14.285\%$ for $0 < j \leq 5$.

For $n > 1$, the subnet residence time density function and its Laplace transform for an n -layer subnet are

$$f_s(t) = \sum_{k=1}^{\infty} \sum_{y=0}^{2n-3} \sum_{j=0}^{2n-3} q_{(n-1,y)} p_{k,(n-1,y)(n,j)} f_s^{(k)}(t). \quad (4)$$

$$f_s^*(a) = \sum_{k=1}^{\infty} \sum_{y=0}^{2n-3} \sum_{j=0}^{2n-3} q_{(n-1,y)} p_{k,(n-1,y)(n,j)} [f_p^*(a)]^k. \quad (5)$$

From the moment generation property, the expected subnet residence time is

$$E[t_s] = (-1) \frac{df_s^*(a)}{da} \Big|_{a=0} = \sum_{k=1}^{\infty} \sum_{y=0}^{2n-3} \sum_{j=0}^{2n-3} q_{(n-1,y)} p_{k,(n-1,y)(n,j)} y k, \quad (6)$$

where

$$y_k = -k [f_p^*(0)]^{k-1} \frac{df_p^*(a)}{da} \Big|_{a=0} \quad (7)$$

We assume that the AP area residence time of an MN has a Gamma distribution with mean $1/\lambda_p$ ($=E[t_p]$) and variance ν . The Gamma distribution is selected for its flexibility and generality. The Laplace transform of a Gamma distribution is

$$f_p^*(a) = \left(\frac{\gamma \lambda_p}{a + \gamma \lambda_p} \right)^\gamma, \quad \text{where } \gamma = \frac{1}{\nu \lambda_p^2}. \quad (8)$$

For an MN, in the end, the probabilities $\Pi_p(K)$ and $\Pi_s(W)$ that the MN moves across K AP areas and W subnets during a session duration, can be derived as follows [8]:

$$\Pi_p(K) = \begin{cases} 1 - \frac{E[t_o]}{E[t_p]} (1 - f_p^*(\frac{1}{E[t_o]})) & , K = 0 \\ \frac{E[t_o]}{E[t_p]} (1 - f_p^*(\frac{1}{E[t_o]}))^2 (f_p^*(\frac{1}{E[t_o]}))^{K-1} & , K > 0 \end{cases} \quad (9)$$

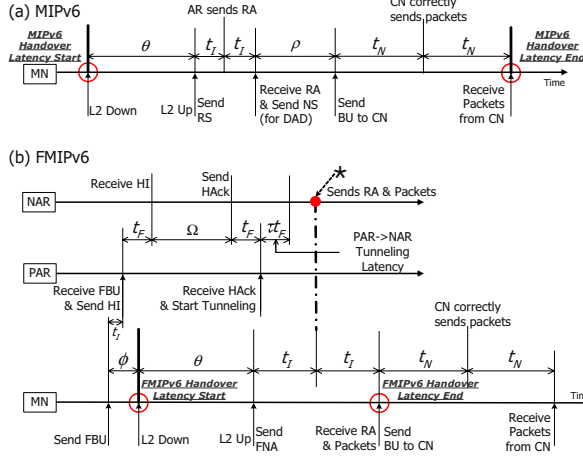


Fig. 4. Timing diagram of handover procedures

$$\Pi_s(W) = \begin{cases} 1 - \frac{E[t_o]}{E[t_s]}(1 - f_s^*(\frac{1}{E[t_o]})) & , W = 0 \\ \frac{E[t_o]}{E[t_s]}(1 - f_s^*(\frac{1}{E[t_o]}))^2(f_s^*(\frac{1}{E[t_o]}))^{W-1}, W > 0 \end{cases} \quad (10)$$

3.2 Handover Latency

In this section, we present analytic handover latency functions to compare our proposal with the existing protocols. The following distance parameters are assumed:

- a: #hops between HA and Border Gateway.
- b: #hops between CN and Border Gateway.
- c: #hops between Border Gateway and AR.
- d: #hops between AR and AP.
- e: #hops between two ARs.

Although all signaling messages defined MIPv6 and FMIPv6 have different bytes of their sizes, for simplicity, we assume that they all have the same delays of delivery if their destination is the same. Let us assume that η is the packet delivery delay in wireless path between AP and MN, and ϵ is the packet delivery delay per one hop in wired path. We define θ as the delay of MN's switching between two APs. Also, ρ and Ω are defined as the RFC 2462 DAD latency and the address confirmation latency in FMIPv6, respectively. No specific address confirmation scheme is described in the current FMIPv6, so RFC 2462 DAD is also used for the confirmation scheme ($\rho = 1000ms$) and Ω is 1000ms in our FMIPv6 performance evaluation, too. Lastly, we define τ as the additional weight of packet tunneling and select 1.2 as its default value.

Fig. 4 (a) and 4 (b) show each timing diagram of handover procedures in MIPv6 and FMIPv6, respectively. In this figure, $t_I (= \eta + \epsilon d)$ represents the

delivery delay between MN and AR, $t_N (= \eta + \epsilon(b + c + d))$ delivery delay between MN and CN, and $t_F (= \epsilon e)$ delivery delay between two ARs. Also, ϕ is the delay between the time to send FBU and the time of disconnection (L2-DOWN) with the current AP.

Using such parameters, for the basic MIPv6, the total handover latency per a session duration is defined as follows:

$$\sigma_{MIPv6} = \sum_{x=0}^{\infty} \left(x \Pi_p(x) \theta + x \Pi_s(x) (2t_I + \rho + 2t_N) \right). \quad (11)$$

From the above function with $\rho = 0$, we can get the handover latency for the enhanced MIPv6 equipped with 'aDAD'.

In FMIPv6, MN sends FBU to PAR prior to disconnection with PAR. At this time, the handover procedure of FMIPv6 is divided into two independent procedures; P_I , the procedure to be executed by MN itself with PAR and NAR, and P_{II} , the procedure to be executed by only both PAR and NAR in order to establish the bidirectional tunnel. The two separated procedures will combine into one when NAR receives FNA from MN after MN's subnet movement (This point is marked as ' \star ' in the Fig. 4 (b)).

We first assume that NAR has already received at least HI from PAR, when it receives FNA from MN. Before the two procedures- P_I and P_{II} -combine into one, the completion time of each procedure is defined respectively as follows:

- $C_{P_I} = \phi + \theta + t_I$.
- $C_{P_{II}} = t_I + (2 + \tau)t_F + \Omega$.

If $C_{P_I} > C_{P_{II}}$, NAR has buffered the packets tunneled from PAR and forwards them to MN when it receives FNA. Otherwise, NAR is running its NCoA confirmation procedure or is waiting the packets which will be tunneled from PAR when it receives FNA. At the latter case, NAR should wait the completion of the confirmation procedure or the tunneled packets before it responds to FNA.

Let us assume that the NCoA confirmation procedure of the basic FMIPv6 always yields the successful result (that is, the prospective NCoA is unique). The enhanced FMIPv6 equipped with 'aDAD', by nature, yields the successful result. For the basic FMIPv6, the total handover latency per session is defined as follows:

$$\sigma_{FMIPv6} = \sum_{x=0}^{\infty} \left(x \Pi_p(x) \theta + x \Pi_s(x) (\text{MAX}\{C_{P_I}, C_{P_{II}}\} + t_I - \theta - \phi) \right). \quad (12)$$

We can get the handover latency for the enhanced FMIPv6 equipped with 'aDAD' from the above function with $\Omega = 0$ in $C_{P_{II}}$.

3.3 Numerical Results

Now, we examine the total handover latency per session, where each protocol is employed. For all examinations, the following fixed parameters are used: $\lambda_o = 0.0033$ (a session time is 300 seconds.), $\epsilon = 0.005$ second, $\nu = 1.0$, $a = 10$, $b =$

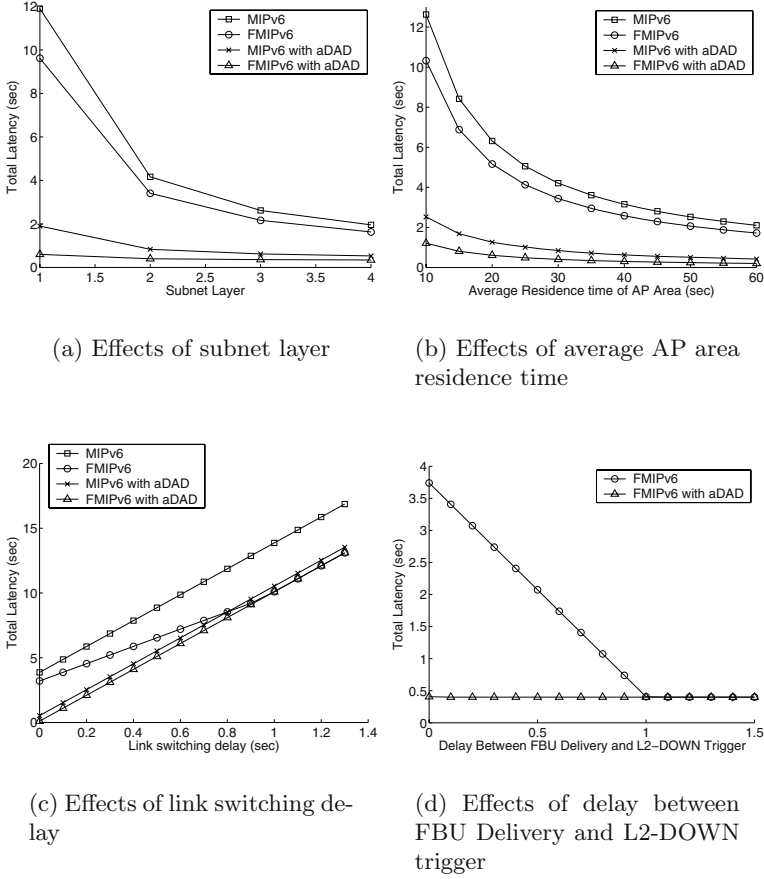


Fig. 5. Total handover latency per a session duration (=300 sec.)

$7, c = 3, d = 1, e = 2$, and $\eta = 0.01$ second. As the target of investigation, we select the following four changeable parameters and their default values: $n = 2$ (subnet layer is 2), $\lambda_p = 0.033$ (that is, the mean of AP area residence time is 30 seconds), $\theta = 0.03$ second, and $\phi = 0.1$ second.

Fig. 5 diversely depicts the total handover latency per session with respect to each changeable parameter. From the figure, we can know that MIPv6 and FMIPv6 handover latency is considerably reduced when our scheme is applied into them. Fig. 5 (a) depicts the total handover latency of each protocol with respect to the subnet layer. It shows that the reduction of latency becomes high when a subnet contains many AP areas. The total handover latency depends mostly on Layer 2 switching delay, if there are many AP areas in a subnet; however, MIPv6 and FMIPv6 equipped with 'aDAD' are under little influence of such system deployment. Fig. 5 (b) shows that the handover process occupies much time within the whole session duration when MN moves across AP areas and subnets more frequently.

Fig. 5 (c) reveals the relationship between the handover latency and the delay of link switching between two APs. When the switching delay becomes high, all protocols' handover latencies become high, too. We can also see that the basic FMIPv6's handover latency becomes equal to that of the FMIPv6 equipped with 'aDAD' when the link switching delay is above 0.9 seconds. That is, the procedure P_I becomes the dominant factor of handover latency. In such case, therefore, the address confirmation time in FMIPv6 does not affect the whole handover latency. Fig. 5 (d) shows that FMIPv6's handover latency becomes low if an MN sends an FBU to the PAR more early, before it disconnects with the PAR. If the FBU can be delivered to the PAR as soon as possible, an NAR also receives an HI early in FMIPv6 handover process. In such case, therefore, the NAR can have enough time to confirm the prospective NCoA.

4 Conclusions

In this paper, we have shown that the address configuration and confirmation procedure occupy the most part of MIPv6 and FMIPv6 handover latency. From all examinations with diverse parameters, the results have provided that 80% ~ 85% of whole handover latency is taken by the address configuration and confirmation process. Particularly, the 'predictive mode' operation should be executed rather than 'reactive mode' operation in FMIPv6 to reduce the handover latency and the packet loss. The address configuration and confirmation process, which is the time consuming process and occupy the most part of the handover latency, should be improved. Therefore, the new scheme named 'Advanced DAD' is designed as a substitute of RFC 2462 DAD procedure used currently in MIPv6 and FMIPv6 for the seamless IP handover in this paper. In this scheme an AR executes the address configuration and confirmation procedure as a background process, and it creates and stores duplication-free addresses in a cache. When a handover is executed, the AR just allocates one of these addresses to an MN. The major benefits of our scheme are to completely eliminate the CoA configuration and confirmation latency involved in any seamless handover schemes and make no address collision occur provided there is no packet loss.

References

1. D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," draft-ietf-mobileip-ipv6-24.txt, June 2003.
2. S.-J. Vaughan-Nichols, "Mobile IPv6 and The Future of Wireless Internet Access," IEEE Computer, Vol. 36, Issue 2, pp. 18–20, February 2003.
3. Thomson, S. and T. Narten, "IPv6 Stateless Address Autoconfiguration," IETF RFC 2462, December 1998.
4. R. Koodli, "Fast Handovers for Mobile IPv6," draft-ietf-mobileip-fast-mipv6-07.txt, IETF, September 2003.
5. T. Nartan and R. Draves, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," IETF RFC 3041, January 2001.

6. I. F. Akyildiz, Y. B. Lin, W. R. Lai, and R. J. Chen, "A New Random Walk Model for PCS Networks," IEEE JSAC, Vol.18, No.7, pp.1254–1260, July 2000.
7. Y.H. Han, *Hierarchical Location Chacing Scheme for Mobility Management*, Ph.D. dissertation, Dept. of Computer Science and Engineering, Korea University, December 2001.
8. Y. B. Lin, "Reducing Location Update Cost in A PCS Network," IEEE/ACM Transactions on Networking, Vol.5, No.1, pp. 25–33, Febraury 1997.
9. N. Moore, "Optimistic Duplicate Address Detection," draft-moore-ipv6-optimistic-dad-03.txt, IETF, September 2003.
10. S.H. Hwang, B.K. Lee, Y.H. Han, and C.S. Hwang, "An adaptive hierarchical mobile IPv6 with route optimization," Proc. of the 57th IEEE VTC 2003-Spring, Vol. 3, pp. 1502–1506, April, 2003

TCP Optimization through FEC, ARQ, and Transmission Power Tradeoffs^{*}

Dhiman Barman¹, Ibrahim Matta¹, Eitan Altman², and Rachid El Azouzi³

¹ 111 Cummington St., Dept. of Computer Science, Boston University, Boston, MA 02215, USA {dhiman,matta}@cs.bu.edu,

² INRIA, B.P. 93, 2004 Route des Lucioles, 06902, Sophia-Antipolis Cedex, France altman@sophia.inria.fr

³ LIA/CERI-Université d'avignon Agroparc 339, chemin des Meinajaries BP 1228, 84911 Avignon, France Rachid.Elazouzi@lia.univ-avignon.fr

Abstract. TCP performance degrades when end-to-end connections extend over wireless connections — links which are characterized by high bit error rate and intermittent connectivity. Such link characteristics can significantly degrade TCP performance as the TCP sender assumes wireless losses to be congestion losses resulting in unnecessary congestion control actions. Link errors can be reduced by increasing transmission power, code redundancy (FEC) or number of retransmissions (ARQ). But increasing power costs resources, increasing code redundancy reduces available channel bandwidth and increasing persistency increases end-to-end delay. The paper proposes a TCP optimization through proper tuning of power management, FEC and ARQ in wireless environments (WLAN and WWAN). In particular, we conduct analytical and numerical analysis taking into account the three aforementioned factors, and evaluate TCP (and “wireless-aware” TCP) performance under different settings. Our results show that increasing power, redundancy and/or retransmission levels always improves TCP performance by reducing link-layer losses. However, such improvements are often associated with cost and arbitrary improvement cannot be realized without paying a lot in return. It is therefore important to consider some kind of net utility function that should be optimized, thus maximizing throughput at the least possible cost.

1 Introduction

When TCP connections extend over wireless links, many factors such as interference, multipath fading, user mobility and atmospheric conditions may cause errors resulting in frame losses over the wireless links. It's not fair to expect TCP to perform well over such links because it was designed to perform mainly over wired networks where there is almost no channel or random packet losses. Nor can we replace TCP overnight because 90% of Internet applications exchange traffic through this protocol.

^{*} This work was supported in part by NSF grants ANI-0095988, ANI-9986397, EIA-0202067 and ITR ANI-0205294, and by grants from Sprint Labs and Motorola Labs.

Various strategies have been proposed to combat this problem which can be classified along the following strategies: split-connection, proxy-based, link-layer error control, and end-to-end.

In the split-connection approach such as [3], the wireless link is hidden from the TCP sender by terminating the connection at the base station. A more reliable connection is established over the wireless link. But this approach violates the end-to-end semantics of TCP and there exists overhead of maintaining state information at the base station.

In proxy-based approaches such as Snoop [4], a base station monitoring a wireless link tries to do local recovery for wireless losses. In ELN-based versions, a TCP-aware base station monitoring the packets going over the wireless link marks the acknowledgment packets with ELN (Explicit Loss Notification) bit if packet loss is due to wireless loss and not congestion. TCP performance can thus be improved at the cost of extra state maintenance at the base station.

Link-layer approaches such as forward error correction (FEC) and automatic repeat request (ARQ) attempt to hide channel losses from the TCP sender by cleaning wireless links of the wireless losses. Link-layer solutions are appealing as they do not incur the overhead associated with TCP-awareness. Introduction of FEC consumes wireless resources but at the same time reduces the link loss rate. On the other hand, link loss rate can be reduced by increasing the transmission power but this means higher transmission costs. Link losses can also be alleviated by using retransmission mechanisms such as ARQ schemes but they increase end-to-end delay thus reducing end-to-end throughput. In [5], the authors analyzed the tradeoff between the bandwidth utilized by FEC and goodput gained by a TCP session. They proposed an algorithm to compute the optimal code that maximizes TCP goodput. In our model, wireless channels are more vulnerable to bit error rates than erasures (i.e., frames getting lost in the wireless channel).

In the end-to-end approach, attempts are made to improve TCP performance at end hosts without any aid from the network (e.g. from base stations). Such proposals include TCP-SACK and TCP-Westwood [7].

Two main conclusions can be drawn from the previous schemes, either we should hide the wireless link from the TCP sender or we should make the TCP sender aware of the occurrence of wireless loss. In this paper, we study the joint effects of FEC, ARQ-SR (Selective Repeat) and power management on TCP performance with and without ELN-type feedback. We evaluate the effects of wireless link losses and the corrective measures on the end-to-end behavior of TCP using different models and settings. In a recent work [12], the authors studied a combined effect of power and FEC on TCP performance and they provided analytical expressions for optimal values of power and redundancy.

Our contribution in this paper is four fold. Firstly, we consider all the significant parameters (transmission power, FEC and ARQ) for evaluating TCP performance over wired/wireless links. Secondly, we consider two different models of link-layer mechanisms and show the effects on the end-to-end measured RTT of TCP. Thirdly, we consider end-to-end TCP performance under settings in which there are both congestion and wireless losses. Finally, we apply the

analysis on an informed model of TCP (which does not take congestion control actions on wireless losses) and show the combined effects of the aforementioned parameters on TCP. Our analytical and numerical evaluations show that increasing power, redundancy and/or retransmission levels always improves TCP performance by reducing link-layer losses. However, such improvements are often associated with cost and arbitrary improvement cannot be realized without paying a lot in return. It is therefore important to consider some kind of net utility function that should be optimized, thus maximizing throughput at the least possible cost.

The remainder of the paper is organized as follows. In Section 2, we define the model of TCP in a hybrid wired/wireless environment. We describe the model for FEC (Section 2.1), ARQ (Section 2.2), and power management (Section 2.3). In Section 3 we derive TCP throughput in the context of our model. We also show expected RTT computation under different link-layer models (Section 3.2). In Section 4, we consider an end-to-end model for informed TCP throughput in which the TCP sender knows about wireless losses. We present our numerical evaluation in Section 5. Finally we conclude and describe future avenues for research in Section 6.

2 Model

2.1 Forward Error Correction

In our model (as represented in Figure 1), we assume that the TCP packets of size MSS traverse both wired and wireless links. We consider that the base station has large enough buffer. This was found to prevent unfairness among TCP flows caused by buffer size smaller than advertised windows [15]. Even if buffer overflow occurs at the base station, our model effectively treats such losses as congestion. It is assumed that packets are acknowledged and acknowledgments traverse through the same base station [9].

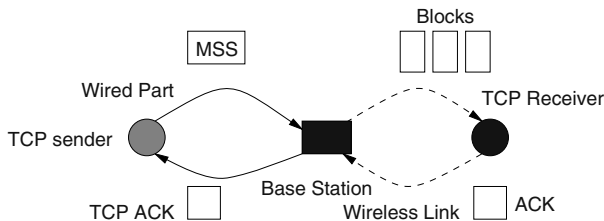


Fig. 1. A model for a hybrid wired/wireless network. A TCP connection extends over a wireless link through a base station. MSS is the packet size on the wired part and a packet is chunked into smaller blocks on the wireless link

We consider a wireless link where data are transmitted as blocks of length K each. FEC encodes each one of these blocks into a codeword of length N ,

with $N > K$. The *redundancy ratio*, x , is defined as the ratio of the amount of redundancy due to FEC, $(N - K)$, to the block length, K , i.e.,

$$x = (N - K)/K \quad (1)$$

We denote by B the bandwidth of the wireless channel, and by D its one-way propagation delay. Each MSS sized segment is divided into $X = \frac{MSS}{K}$ many K -sized blocks. A TCP segment is decoded properly if all X blocks are received uncorrupted and decoded by the receiver. The link layer implements ARQ-SR/FEC error control scheme. We assume strong CRC code so that the probability of not detecting a corrupted block is practically zero. In contrast, only a subset of bit errors can be corrected by FEC [13]. We also assume that the feedback (ACK/NACK) messages are well protected or the probability of acknowledgement losses is negligible so that no retransmissions are needed for these messages.

The underlying hybrid ARQ-SR/FEC mechanism is characterized by (N, K, δ, e_c, e_d) , where N is the number of bits in a code block, K is the number of information bits in a code block, δ is the maximum number of allowable retransmissions, e_c is the maximum number of correctable bits in a code block, and e_d is the maximum number of corrupted bits which can be detected. In here, a block corresponds to a fixed-size link-layer packet. Note that N depends on K, e_c and e_d . In general, an (N, K) code with minimum distance d_{min} can detect e_d errors and correct e_c errors, where $e_d + e_c \leq d_{min} - 1$ and $e_c \leq e_d$ [13]. The coding gain is given by $G_{coding} = \frac{K}{N} \cdot d_{min}$. For example, if Reed-Solomon encoding is used, then $G_{coding} = \frac{1}{1+x} \cdot (N - K + 1) = \frac{K \cdot x + 1}{1+x}$ using Eq. (1).

2.2 Automatic Repeat Request (Selective Request)

If FEC does not succeed to decode one block, the link-level error mechanism turns to ARQ-SR for the retransmission of the block. The retransmission will be attempted a maximum number of times, referred to as the persistency of ARQ-SR and denoted by δ , $\delta = 0, 1, 2, \dots$. $\delta = 0$ means that there are no retransmissions and that ARQ-SR is disabled. If after δ trials the frame does not get through the wireless link, ARQ-SR assumes that the frame cannot be locally recovered, and leaves for TCP the correction of the frame on an end-to-end basis. This is important because if the link layer keeps trying indefinitely, it would increase the end-to-end delay and RTT and may lead to TCP timeout (unless the base station informs the TCP sender about the local recovery process.)

The ARQ-SR receiver at the output of the wireless link acknowledges each block with a positive ACK or a NACK. When a NACK is received at the input of the wireless link, the corresponding block is directly retransmitted, and given priority over all other blocks that have not been transmitted. We have considered two different models of ARQ-SR in computing average end-to-end RTT which we describe in detail in Section 3.2.

2.3 Power Management

Link reliability can be improved by increasing the transmission power. In fact, the bit error probability, p_e , decreases when the ratio (E_b/N_0) increases, where E_b is the received energy per bit and N_0 is the noise power spectral density. Note that the relationship between the bit error probability, and the ratio (E_b/N_0) is a function of the modulation technique. Let y represent the transmission power. The received energy per bit, E_b is equal to $A \cdot y/B$ where A is the attenuation. Increasing the transmission power improves TCP performance but causes greater energy consumption and aggravates the interference with other neighboring communications.

3 The Analytical Framework

3.1 TCP Throughput Evaluation

Different analytical formulas for TCP throughput have been proposed in the literature. The general form can be written as follows:

$$\lambda(x, y, \delta) = \frac{1}{1+x} \cdot f(RTT, P_{Loss}) \quad (2)$$

where $f(\cdot, \cdot)$ is the TCP throughput function which depends on RTT and average packet loss probability. The most commonly used formula for TCP throughput is given by [14]:

$$f(RTT, p) = \frac{1}{RTT} \min \left\{ W_{\max}, \frac{1}{\sqrt{\frac{2bp}{3}} + T_0 \min(1, 3\sqrt{\frac{3bp}{8}})p(1+32p^2)} \right\} \quad (3)$$

where W_{\max} is the maximum congestion window size of the TCP sender, b represents the effect of delayed acknowledgements, and T_0 is the TCP retransmission timeout value. RTT is the round trip time which depends on the persistency level of ARQ-SR, δ , the amount of redundancy, x , and packet loss probability, P_{Loss} . P_{Loss} is the probability that a TCP segment is discarded because of link errors in the wireless channel. The probability P_{Loss} can be evaluated as:

$$P_{Loss} = [1 - (1 - P_{Block})^X] \quad (4)$$

where P_{Block} is the block error probability at the output of the decoder. P_{Block} depends on x , δ , and y . We consider Reed-Solomon coding and approximate $P_{Block} \approx [1 - (1 - p_e)^K]^{\delta+1}$. p_e depends on power and the particular modulation scheme as tabulated in Table 1. In our numerical evaluation we consider three modulation schemes, namely, Gaussian M-ary Shift Keying (GMSK), Differentiated Binary Phase Shift Keying (DBPSK) and Gaussian Frequency Shift Keying (GFSK).¹

¹ It is to be noted that there is an effective method for dealing with correlated error channels by using interleaved coded data such that the bursty channel is transformed into a channel having independent errors [13].

Table 1. p_e for different modulation schemes, $\epsilon \leq (2^K - 1)[4p_e(1 - p_e)]^{d_{min}/2}$, probability of detecting errors in a block is greater than $1 - \epsilon$, α is a constant, A is the attenuation, N_0 is the noise spectral density, ΔF is the size of the frequency band, and the factor $\frac{1+Kx}{1+x}$ is the coding gain for Reed-Solomon coding.

Modulation	GMSK	DBPSK	GFSK
p_e	$\frac{1}{2} \operatorname{erfc}\left(\sqrt{\frac{\alpha A y}{N_0 \Delta F} \frac{Kx+1}{1+x}}\right)$	$\frac{1}{2} \exp\left(-\sqrt{\frac{\alpha A y}{N_0 \Delta F} \frac{Kx+1}{1+x}}\right)$	$\frac{1}{2} \exp\left(-\frac{1}{2} \sqrt{\frac{\alpha A y}{N_0 \Delta F} \frac{Kx+1}{1+x}}\right)$

3.2 RTT Computation

In this section, we compute the average RTT of a connection under two models of link-layer transmissions. We assume that data blocks are quickly acknowledged by ARQ-SR and sizes of acknowledgements are of negligible size as compared to data blocks. Thus the transmission of a block and reception of its acknowledgement takes $\tau = 2D + \frac{N}{B}$. Let $\delta_i \in \{0, 1, \dots, \delta\}$, $i = 1, 2, \dots, X$ be the number of times we retransmit block i of a TCP packet (recall a TCP packet of size MSS is divided into X blocks each of size K).

Model I: In the first model of expected RTT computation, we assume that the next block is transmitted after the sender receives acknowledgement of the previous block (i.e., stop-and-wait). Then, the round-trip time of a TCP packet can be written as:

$$RTT = T + 2D + \frac{XN}{B} + \sum_{i=0}^X \delta_i \tau \quad (5)$$

where T is the round trip time of the wired part of the TCP connection. RTT is a random variable and the randomness comes from random variables δ_i , which are i.i.d. and geometric. In order to capture the effect of the wireless part, we are assuming that delay variability on the wired part is less [8]. We have the following:

$$P\{\delta_i = k\} = \begin{cases} \frac{P_{Block}^k (1 - P_{Block})}{1 - P_{Block}^{\delta+1}} & 0 \leq k \leq \delta, \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

The expected RTT, $E[RTT]$ is derived in the full version of the paper [6] and is given by $E[RTT] = T + 2D + \frac{XN}{B} + X\tau P_{Block} \left[\frac{1}{1 - P_{Block}} - \frac{(\delta+1)P_{Block}^\delta}{1 - P_{Block}^{\delta+1}} \right]$. We assume that retransmission is done with probability 1.

Model II: In this model, we assume that all the blocks are transmitted back-to-back, then it will lead to a different value of $E[RTT]$ (derived in the full version) given by, $E[RTT] = T + 2D + \frac{XN}{B} + \tau\delta - \frac{1}{(1 - P_{Block}^{\delta+1})^X} \int_0^{\frac{XN}{B} + \tau\delta} \prod_{i=1}^X \left(1 - P_{Block}^{\lfloor (z - \frac{iN}{B})/\tau \rfloor + 1}\right) dz$. The expression for $E[RTT]$ is difficult to solve analytically, and therefore, we resort to evaluating it numerically.

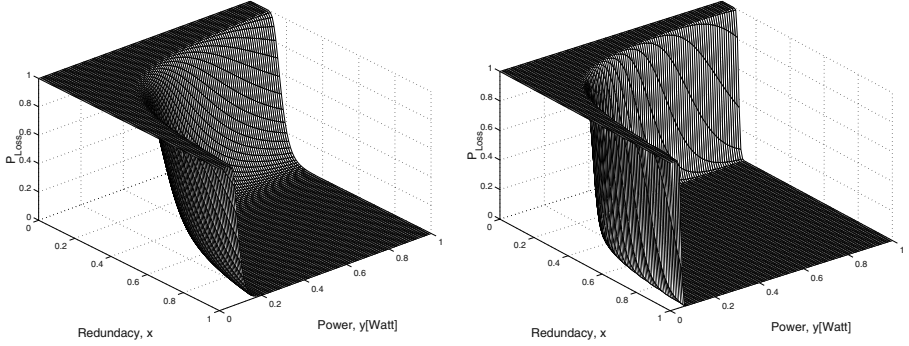


Fig. 2. Behavior of the segment loss probability, P_{Loss} , vs. the redundancy, x , the transmission power, y and the persistency level, $\delta = 0$ (left) and $\delta = 10$ (right)

3.3 Cost Evaluation

Consider a need to transfer S TCP segments, each MSS size segment will be transmitted in a total X of $K(1+x)$ sized codewords over the wireless link, where each codeword could be retransmitted $0 \leq \delta_i \leq \delta$ times. Thus the cost of transfer depends on the transmission power, the amount of redundancy introduced and the level of persistency. We consider two cost terms: a term which takes energy consumption into account, and a term which considers the amount of wireless resources employed. If the redundancy introduced by FEC is x , then $[S \cdot MSS \cdot (1+x)](1+E[\delta_i])$ bits must be transmitted in order to deliver S segments of MSS bits each. Accordingly, the cost of the resources required to complete the transfer is given by $c_{resources} = k_r \cdot S \cdot MSS \cdot (1+x)(1+E[\delta_i])$ where k_r (expressed in bit^{-1}) is a constant which represents the cost of the bandwidth required to transfer a bit. Given that the energy transmitted per bit is given by $\frac{y}{B}$, the energy consumption required to complete the transfer is $[S \cdot MSS \cdot (1+x)](1+E[\delta_i])y/B$. As a result, if k_e (expressed in Joule^{-1}) represents the cost of a unit of energy, then the total cost is $c_{energy} = k_e \cdot S \cdot MSS \cdot (1+x)(1+E[\delta_i]) \cdot y/B$. Accordingly, when the redundancy is x , the energy transmitted per bit is y and the persistency is bounded by δ , the cost of the transfer can be evaluated as:

$$c(x, y, \delta) = c_{energy} + c_{resources} = S \cdot MSS \cdot (1+x)(1+E[\delta_i]) \left(\frac{k_e y}{B} + k_r \right) \quad (7)$$

The constants k_r and k_e depend on many factors such as user preferences, the type of terminal and the costs of power and communication. There are other cost functions that could be realized based on the same metrics.

3.4 Objective Function Maximization

Let us define the objective function $\gamma(x, y, \delta)$ as follows:

$$\gamma(x, y, \delta) = \lambda(x, y, \delta) / c(x, y, \delta) \quad (8)$$

Now we want to evaluate the values of x, y and δ which maximize the function $\gamma(x, y, \delta)$ given by:

$$\gamma(x, y, \delta) = \frac{f(E[RTT], P_{Loss})}{(1+x)^2 S \cdot MSS \cdot (1+E[\delta_i])(k_e y/B + k_r)} \quad (9)$$

where Eq. (9) is obtained by substituting Eq. (2) and (7) in Eq. (8).

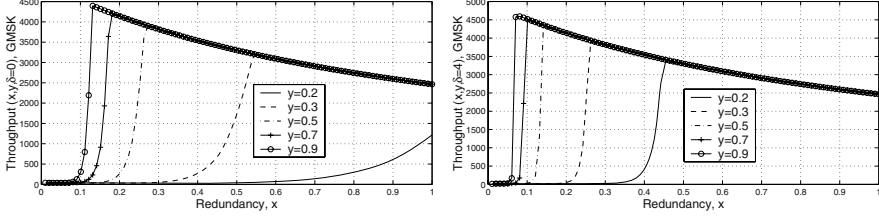


Fig. 3. Behavior of the throughput vs. the redundancy ratio for different values of the transmission power, y and persistency level, $\delta = 0$ (left) and $\delta = 4$ (right) for RTT Model I

4 End-to-End TCP Performance

Using Eq. (3) we may not know the ideal desired behavior of TCP which discriminates congestion losses from channel losses. In this section, we consider a channel error informed TCP throughput model which shows the performance of informed TCP under different conditions. This throughput model should reflect the desired TCP behavior over lossy links, i.e., i) the TCP source should not reduce its sending rate in case of packet loss due to wireless link error (in fact it should keep probing for available bandwidth), and ii) the source should follow normal TCP control rules otherwise. We follow the analysis shown in [11] to model an Internet path having wired and wireless parts. Although the work modeled an end-to-end path using a four-state Markov model, in the analysis the authors only considered average case losses without accounting for bursty losses due to correlated channel conditions. It is part of our future work to derive throughput formula for *informed* TCP using a more detailed model that explicitly account for bursty channel losses similar to the existing work studying standard TCP behavior [1,2,10]. Using the analytical model of the desired behavior derived in [11], we have:

$$\lambda(x, y, \delta) = \frac{1}{4E[RTT]} \left(3 + \sqrt{25 + 24\theta} \right), \quad \theta = \frac{1 - P(S_1)}{P_{e2e} - P(S_1)} \quad (10)$$

Note that $\theta \approx \frac{1 - P_{Loss}}{P(I_1)}$ where $P(S_1)$ is the packet loss probability on the wireless link given by Eq. (4), and $P(I_1)$ is average congestion loss probability (we set it

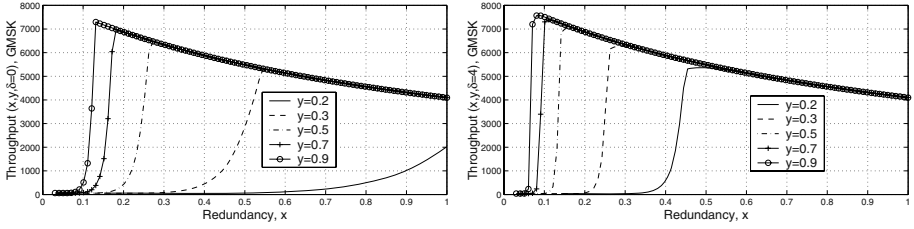


Fig. 4. Behavior of the throughput vs. the redundancy ratio for different values of the transmission power, y and persistency level, $\delta = 0$ (left) and $\delta = 4$ (right) for RTT Model II

to 1% in our experiments). We obtain a utility function by replacing Eq. (10) as $f(\cdot, \cdot)$ in Eq. (9) and then maximize it. Note that Eq. (10) reduces approximately to the throughput equation (without timeouts) given in [14] when $P(S_1) = 0$.

5 Numerical Analysis

In this section we apply the analytical framework developed in the previous sections to the Gaussian M-ary Shift Keying (GMSK) modulation technique, which is used in the General Packet Radio Service. The behavior of Differentiated Binary Phase Shift Keying (DBPSK), which is used in IEEE 802.11 is found similar to GMSK. We present the plots corresponding to DBPSK in the full version.

Table 2. Parameters used in numerical examples. A (attenuation) is given by $\left(\frac{c}{4\pi df_c}\right)^2$. W_{max} is set to > 64 packets. $c = 3 \times 10^8$ m/s, $\alpha = 0.8$, $b = 1$, $\epsilon \leq (2^K - 1)[4p_e(1 - p_e)]^{d_{min}/2}$

	K	MSS	ΔF	$N_0(e^{-20})$	$k_e/k_r, k_r$	$d(m)$	T_0	$T(ms)$	D	f_c
GMSK	260 bits	128 bytes	25MHz	1.379	100,1	150	4s	100	10ms	500MHz

In our experiments we have used a set of values for the parameters which we tabulate in Table 2. In Figure 2, we observe the variation of TCP packet loss probability, P_{Loss} for $\delta = 0$ and $\delta = 10$. The value of P_{Loss} decreases with increasing redundancy, x and increasing power level, y (cf. Eq.(4)). More redundancy increases the error correcting capability of the codeword and more power increases the signal-to-noise ratio reducing the bit error probability. In Figure 3, we observe the variation of TCP end-to-end throughput using Model I of the expected RTT. We can see that throughput increases with increase in power level. For a given value of power and δ , with increase in x , throughput first increases to a maximum value and then reduces. The throughput reduces because

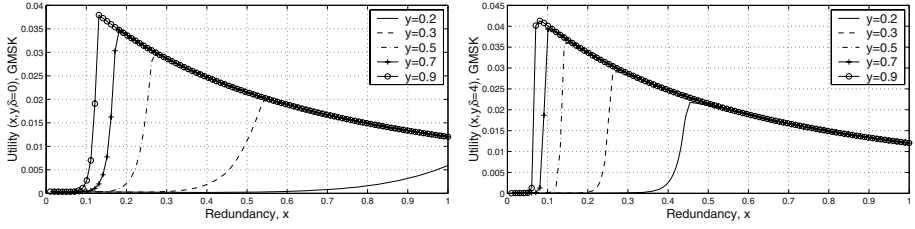


Fig. 5. Behavior of the utility vs. the redundancy ratio for different values of the transmission power, y and persistency level, $\delta = 0$ (left) and $\delta = 4$ (right) for RTT Model I

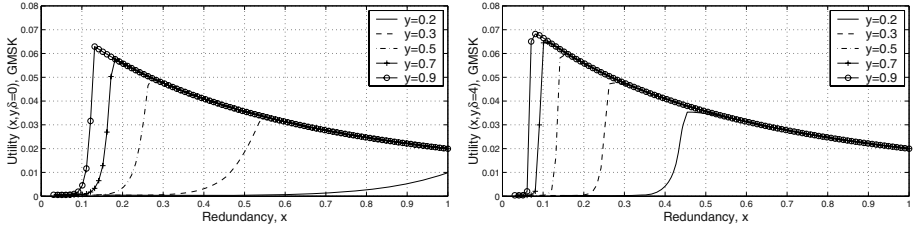


Fig. 6. Behavior of the utility vs. the redundancy ratio for different values of the transmission power, y and persistency level, $\delta = 0$ (left) and $\delta = 4$ (right) (RTT Model II)

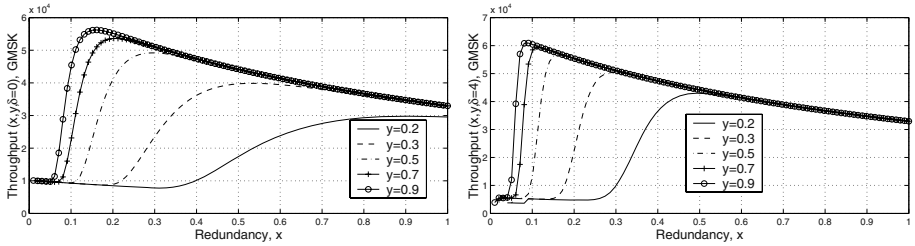


Fig. 7. Behavior of the throughput vs. the redundancy ratio for different values of the transmission power, y and persistency level, $\delta = 1$ (left) and $\delta = 4$ (right) (wireless-loss informed TCP model)

more redundancy reduces the effective bandwidth. We can see that increase in δ for a given power level and x , improves the throughput. We observe similar behavior in Figure 4 with RTT model II. In Figure 5, we observe the interesting variation of utility function depending on power, redundancy and persistency using RTT model I. For a given value of power and persistency level, the utility function attains a maximum value at a certain redundancy level. Beyond that, utility reduces after a point because the gain in throughput is outweighed by the increased cost of redundancy. We observe the same interesting pattern of utility function variation in Figure 6 as we observe in Figure 5.

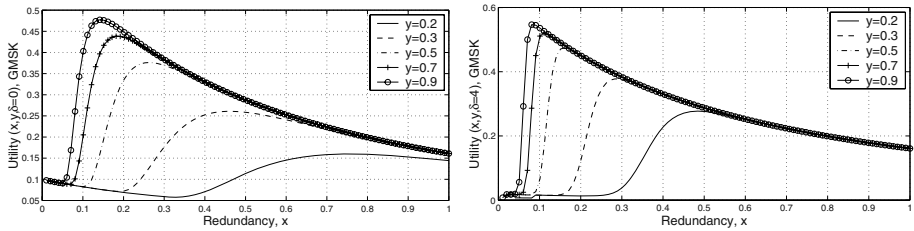


Fig. 8. Behavior of the utility vs. the redundancy ratio for different values of the transmission power, y and persistency level, $\delta = 1$ (left) and $\delta = 4$ (right) (wireless-loss informed TCP model)

In Figure 7, we plot the throughput using the wireless-loss informed TCP throughput model found in Eq. (10). We kept the end-to-end congestion loss rate at 1% and maximum window size is set to > 64 packets. We observe the improved TCP performance as a result of TCP's awareness of the reason of packet loss. Numerically, we find the optimum values of the parameters, x_{opt} , y_{opt} and δ_{opt} to be 0.0201, 0.9265 and 3, respectively. Figures 2-8 represent the modulation scheme GMSK. We repeated the same experiments for the modulation scheme DBPSK, for which we show representative results in the full version.

6 Conclusion and Future Work

New kinds of losses, additional constraints and new infrastructure demand new solutions to reliable data delivery over wireless channels. In this paper, we have tried to observe a combined effect of all palpable metrics which affect TCP performance in this new environment. We have seen that increasing power, redundancy and retransmission levels always improves the performance by reducing link-layer losses. But such improvements are often associated with cost and arbitrary improvement cannot be realized without paying a lot in return. It is therefore important to consider some kind of net utility function that should be optimized, thus maximizing throughput at the least possible cost. As a part of future work, we would like to obtain some closed-form analytical expressions so that we gain better insights into the interactions among different parameters. Such expressions can be used by the base station to dynamically adjust control parameters according to channel conditions. We intend to validate our analysis using ns-2 simulations.

References

1. Eitan Altman, Kostya Avrachenkov, and Chadi Barakat. TCP in Presence of Bursty Losses. In *Performance Evaluation*, volume 42, pages 129–147, October 2000.

2. F. Anjum and L. Tassiulas. On the Behavior of Different TCP Algorithms over a Wireless channel with Correlated Packet Losses. In *Proceedings of ACM SIGMETRICS*, March 1999.
3. Ajay Bakre and B.R. Badrinath. I-TCP: Indirect TCP for Mobile Hosts. In *Proceedings of the 15th International Conference on Distributed Computing Systems*, 1995.
4. H. Balakrishnan, S. Seshan, and R. Kartz. Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks. In *ACM Wireless Networks*, 1(4), Dec 1995.
5. Chadi Barakat and Eitan Altman. Bandwidth Tradeoff between TCP and Link-level FEC. In *Proceedings of IEEE International Conf. on Networking, France*, July 2001.
6. Dhiman Barman, Ibrahim Matta, Eitan Altman, and Rachid El Azouzi. TCP Optimization through FEC, ARQ and Transmission Power Tradeoffs. Technical report, Boston University, Computer Science Department, Boston, MA 02215, 2003.
7. Antonio Capone and Fabio Martignon. Bandwidth Estimates in the TCP Congestion Control Scheme. In *Proceedings of IWDC'01*, Italy, 2001.
8. Marcelo M. Carvalho and J.J.Garcia-Luna-Aceves. Delay Analysis of IEEE 802.11 in Single-Hop Networks. In *Proceedings of IEEE ICNP*, Atlanta, Georgia, 2003.
9. Mun Choon Chan and Ramachandran Ramjee. TCP/IP Performance over 3G Wireless Links with Rate and Delay Variation. In *Mobicom'02*, 2002.
10. A. Chockalingam, M. Zorzi, and R.R. Rao. Performance of TCP on Wireless Fading Links with Memory. In *Proceedings of IEEE ICC*, June 1998.
11. Özgür B. Akan and Ian F. Akyildiz. ARC: The Analytical Rate Control Scheme for Real-Time Traffic in Wireless Networks. 2003.
12. Laura Galluccio, Giacomo Morabito, and Sergio Palazzo. An Analytical Study of a Tradeoff Between Transmission Power and FEC for TCP Optimization in Wireless Networks. In *Proceedings IEEE INFOCOM'03*, 2003.
13. J.G.Proakis. *Communication Systems Engineering*. Prentice Hall International Editions, 1994.
14. J. Padhye, V. Firoiu, D. Towsley, and J. Kurose. Modeling TCP Throughput: A Simple Model and Its Empirical Validation. In *Proceedings of ACM/SIGCOMM '98*, Vancouver, Canada, October 1998.
15. Saar Pilosof, Ramachandran Ramjee, Danny Raz, Yuval Shavitt, and Prasun Sinha. Understanding TCP Fairness over Wireless LAN. In *Proceedings of IEEE INFOCOM'03*, 2003.

Sliding Mode Congestion Control in Differentiated Service Communication Networks

Hassan Ebrahimirad and M.J. Yazdanpanah

Control and Intelligent Processing Center of Excellence

Department of Electrical and Computer Engineering

University of Tehran, Iran

`h.ebrahimirad@ece.ut.ac.ir, yazdan@ut.ac.ir`

Abstract. The rapid growth of the Internet and increased demand to use the Internet for voice and video applications necessitate the design and utilization of new Internet architectures with effective congestion control algorithms. As a result, the Differentiated Service (Diff-Serv) architectures were proposed to deliver Quality of Service (QoS) in TCP/IP networks. Network congestion control remains a critical and high priority issue, even for the present Internet architecture. The aim of this paper is to design a robust active queue management system to secure high utilization, bounded delay and loss, while the network complies with the demands each traffic class sets. To this end, variable structure control theory is used and a sliding mode congestion controller is designed. Simulation results of the proposed control action demonstrate the effectiveness of the controller in providing robust queue management system.

1 Introduction

Diff-Serv architecture will try to provide QoS by using differentiated services aware congestion control algorithms. Recently several attempts have been made to develop congestion controllers [1-6], mostly using linear control theory. Despite these efforts, the design of congestion controllers whose performance can be analytically established and demonstrated in practice is still a challenging unresolved problem. It should also be mentioned that, even for the present Internet architecture, network congestion control remains a critical and high priority issue, and is unlikely to disappear in the near future. Furthermore, if we consider the current utilization trends, congestion in the Internet maybe come unmanageable unless effective, robust, and efficient methods for congestion control are developed.

In this paper, the traffic of the network is divided into three types: Premium, Ordinary and Best Effort Traffic Services. For very important people, there are VIPs passes. VIP passes get preferential treatment. This category is likened to our premium traffic Service. For ordinary people, there are common passes. To purchase these tickets, people may have to queue to get the best possible seats, and there is no preferential treatment, unless different prices are introduced for better seats. This category may be likened to our Ordinary Traffic Service. For reasons of economy, another pass may be offered, at a discount price for the opportunists at the door. The last (opportunistic, best effort) class will take any available seat, but also at the risk of not attending the event at all, due to a sold out event. This category is likened to our

Best Effort Traffic Service. The Premium Traffic Service may belong to the EF-PHB in Diff-Serv architecture and is designed for applications with stringent delay and loss requirements on per packet basis that can specify upper bounds on their traffic needs and required quality of service. Typical applications include video on demand, audio, video conferencing, etc. The Ordinary Traffic Service may belong to the first class of the AF-PHB in a Diff-Serv architecture. Note that different priorities may be assigned, without complicating greatly the design. The Ordinary Traffic Service is intended for applications that have relaxed delay requirements and allow their rate into the network to be controlled. These Services use any left over capacity from the Premium Traffic. Note that to ensure that bandwidth is leftover from the Premium Traffic Service a minimum bandwidth may be assigned, e.g. by using bandwidth allocation between services or connection admission. Typical applications include web browsing, image retrieval, e-mail, ftp, etc.

Sliding mode control systems (SMCS) have started in Russia by many researches, like Barbashin [7] and Utkin [8,9] as a special class of nonlinear systems. Due to its excellent invariance and robustness properties, Sliding mode control has been developed into a general design method and extended to a wide spectrum of system types including multivariable, large-scale, infinite-dimensional and stochastic systems. The ideas have successfully been applied to problems as diverse as automatic flight control, control of electric motors, chemical process, helicopter stability augmentation systems, space systems and robots [10-14]. In sliding mode control, controller are designed to drive and then constrain the system to lie within a neighborhood of the switching function [15,16]. There are two main advantages of this approach. Firstly, the dynamic behavior of the system may be tailored by the particular choice of switching functions. Secondly, the closed-loop response becomes totally insensitive to a particular class of uncertainty. In addition, the ability to specify performance directly makes sliding mode control attractive from the design perspective. This design approach consists of two components. The first, involves the design of a switching function so that the sliding motion satisfies design specifications. The second is conserved with the selection of a control law, which will make the switching function attractive to the system state. In this paper, we will make use of sliding mode control theory to congestion control in differentiated services networks. Using the proposed robust control action, congestion control in Premium and Ordinary classes is performed. Best effort class is not controlled. Some computer simulations are provided to illustrate the effectiveness of the proposed sliding mode controller.

The paper is organized as follows: Section 2 describes differentiated services Internet architecture. In Section 3 a fluid flow based model for M/M/1 queues is provided. Sliding mode controller design and simulation results of the system with the controller are provided in Section 4. Finally, the paper is concluded in Section 5.

2 Diff-Serv: New Internet Architecture

Since Int-Serv failed to be adopted for widespread use, the Internet Engineering Task Force (IETF) proposed a more evolutionary approach that did not require significant changes to the Internet infrastructure and provided Differentiation of Services (Diff-

Serv) [17]. To accomplish this, Diff-Serv uses the type of service (ToS) field bits in the IP header, which are now renamed as “DS (Differentiated Services) field” [18]. The functions associated with these bits have also been redefined. The main issue of the Diff-Serv approach is how to standardize a simple set of mechanisms for handling packets with different priorities denoted by the DS field in the IP header. Note that, in Diff-Serv approach, packet classification is performed only at the edges of the network, which reduces the operational complexity in the network core, and makes it more scalable.

On the other hand, no specific measures are taken to assure that the priorities would actually relate to the desired QoS when a packet leaves the edge router. Therefore, the standard Diff-Serv architecture provides only rudimentary QoS, without any quantified guarantees (unlike the ATM case for example). Because of the availability of limited number of bits in the DS field, the Diff-Serv Working Group has defined a small set of building blocks, called per-hop behaviors (PHBs), which are used by the routers to deliver a variation of services. They are encoded in the DS field and they specify the forward behavior each packet expects to receive by the individual routers. The two PHBs being standardized are the Expedited Forwarding (EF) [19], and the Assured Forwarding (AF) [20]. The EF PHB specifies a forwarding behavior with a low loss, low latency, low jitter, and assured bandwidth end-to-end service, thus indirectly providing some QoS. In order to ensure that every packet marked with EF receives this service, EF requires from every router to allocate an adequate level of forwarding resources so that the rate of incoming EF packets is always less than or equal to the rate at which the router can forward them. This is done through a Service Level Agreement (SLA) during the connection setup. In order to preserve this property on an end-to-end basis, EF requires traffic shaping and reshaping in the network. Although there is no specific method set for this, it will most probably be a leaky-bucket buffering algorithm. The AF PHB group specifies a forwarding behavior in which packets see a very small amount of loss. The AF PHB group provides delivery of IP packets in four independently forwarded classes. Within each AF class, two or three drop preference levels are used to differentiate flows. The idea behind AF is to preferentially drop best-effort packets and packets non-conforming to contract when there is congestion. By limiting the amount of AF traffic in the network and by managing the best-effort traffic appropriately, routers can then ensure low loss behavior to packets marked with the EF PHB.

In summary, Diff-Serv architecture will try to provide QoS by using Diff-Serv aware congestion control algorithms. Recently, AQM mechanisms (e.g. RED and its variants) have been proposed [17,21] within the framework of the Diff-Serv architecture [27] to preferentially drop non-conforming against conforming packets.

3 Dynamic Network Model

In this section, a state space equation for M/M/1 queue is presented. The model has been extended to consider traffic delays and includes modeling uncertainties then three classes of traffic services are introduced in a Diff-Serv network.

3.1 Fluid Flow Model

A diagram of a sample queue is depicted in Fig.1. Let $x(t)$ be a state variable denoting the ensemble average number in the system in an arbitrary queuing model at time t . Furthermore, let $f_{in}(t)$ and $f_{out}(t)$ be ensemble averages of the flow entering and exiting the system, respectively.

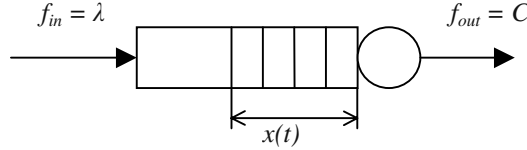


Fig. 1. Diagram of sample queue

$\dot{x}(t) = dx(t)/dt$ can be written as:

$$\dot{x}(t) = f_{in}(t) - f_{out}(t) \quad (1)$$

Equation of this kind of model has been used in the literature, and is commonly referred to as fluid flow equation [22-24]. To use this equation in a queuing system, C and λ have been defined as the queue server capacity and average arrival rate respectively. Assuming that the queue capacity is unlimited, $f_{in}(t)$ is just the arrival rate λ . The flow out of the system, $f_{out}(t)$, can be related to the ensemble average utilization of the queue, $\rho(t)$, by $f_{out}(t) = \rho(t)C$. It is assumed that the utilization of the link, ρ , can be approximated by the function $G(x(t))$, which represents the ensemble average utilization of the link at time t as a function of the state variable. Hence, queue model can be represented by the following nonlinear differential equation [22]:

$$\dot{x}(t) = -CG(x(t)) + \lambda \quad (2)$$

Utilization function, $G(x(t))$, depends on the queuing system under study. If statistical data is available, this function can be empirically formulated. This, however, is not generally the case and $G(x(t))$ is normally determined by matching the results of steady state queuing theory with (2). M/M/1 has been adopted in many communication network traffics. In this model input and service rates both have Poisson distribution function. For M/M/1 the state space equation would be [24]:

$$\dot{x}(t) = -C \frac{x(t)}{1 + x(t)} + \lambda \quad (3)$$

The validity of this model has been verified by a number of researchers [22,25,26,27].

3.2 System Structure and Controller Mechanism

Consider a router of K input and L output ports handling three differentiated traffic classes mentioned above. At each out port a controller has been presented to handle

different classes of traffic flows entering to that port. An instance of the controller is illustrated in Fig. 2. The incoming traffic to the input node includes different classes of traffic. The input node then separates each class according to their class identifier tags and forwards the packets to the proper queue. The output port could transmit packets at maximum rate of C_{server} to destination where

$$C_{server} < C_p + C_r + C_b \quad (4)$$

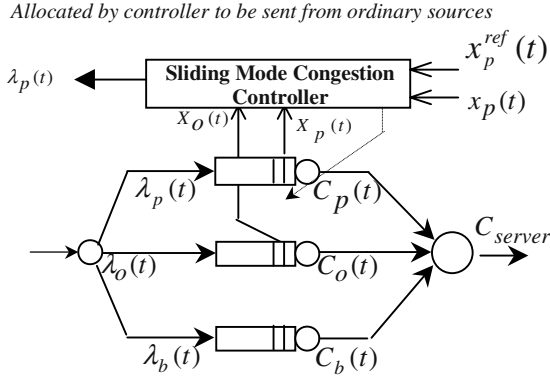


Fig. 2. Control strategy at each switch output port

3.3 Premium Control Strategy

Premium traffic flows needs strict guarantees of delivery. Delay, jitter and packet drops should be kept as small as possible. The queue dynamic model can be as:

$$\dot{x}_p(t) = -C_p(t) \frac{x_p(t)}{1 + x_p(t)} + \lambda_p(t) \quad (5)$$

The control goal here is to determine $C_p(t)$ at any time and for any arrival rate $\lambda_p(t)$ in which the queue length, $x_p(t)$ is kept close to a reference value, $x_p^{ref}(t)$, specified by the operator or designer. So in (5), $x_p(t)$ would be the state to be tracked, $C_p(t)$ is the control signal determined by the congestion controller and $\lambda_p(t)$ is the disturbance.

The goal here is to allocate minimum possible capacity for the premium traffic to save extra capacity for other classes of traffic as well as providing a good QoS for premium flows. Note that we are confined to control signals as

$$0 < C_p(t) < C_{server} \quad (6)$$

In other words the assigned premium capacity must always be less than the maximum server capacity C_{server} . This constraint would make the controller design more difficult.

3.4 Ordinary Control Strategy

In the case of ordinary traffic flows, there is no limitation on delay and we assume that the sources sending ordinary packets over the network are capable to adjust their rates to the value specified by the bottleneck controller. The queue dynamic model is as follows

$$\dot{x}_o(t) = -\frac{x_o(t)}{1+x_o(t)}C_o(t) + \lambda_o(t - \tau) \quad (7)$$

Where, τ denotes the round-trip delay from bottleneck router to ordinary sources and back to the router. The control goal here is to determine $\lambda_o(t)$ at any time and for any allocated capacity $C_o(t)$ so that $x_o(t)$ be close to a reference value $x_o^{ref}(t)$ given by the operator or designer. There are two important points that must be considered, first, $C_o(t)$ is the remaining capacity, $C_o(t) = C_{server} - C_p(t)$ and would be considered as disturbance which could be measured from the premium queue. In our controller scheme we would try to decouple the affect of $C_o(t)$ on the state variable $x_o(t)$, and the another point is that λ_o is limited to a maximum value λ_{max} and no negative λ_o is allowed i.e.

$$0 \leq \lambda_o(t) \leq \lambda_{max} \leq C_{max}$$

3.5 Best-Effort Traffic

As mentioned in the previous section, best effort traffic has the lowest priority and therefore could only use the left capacity not used by Premium and Ordinary traffic flows. So, this class of service is no-controlled.

4 Sliding Mode Congestion Controller Design

In this section, the controller for the congestion control objective is described above. We have made the following assumptions for controller design throughout this paper:

$$C_{max} = 300000 \text{ Packets Per Second}$$

$$\lambda_{max} = 150000 \text{ Packets Per Second}$$

In addition at first is assumed there is not any delay in system ($\tau = 0$). In this system two controllers are designed for premium and ordinary systems. For both premium and ordinary classes, the sliding surface has been selected as follows:

$$s = x(t) - x_{ref}(t) \quad (8)$$

In sliding surface the following condition should be satisfied:

$$\dot{s} = 0 \quad (9)$$

So,

$$\dot{x}(t) - \dot{x}_{ref}(t) = 0 \quad (10)$$

From (3) the main part of control law for premium and ordinary system achieved as follows respectively:

$$C_p(t) = \frac{-\lambda_p + \dot{x}_{ref}(t)}{\left(-\frac{x(t)}{1+x(t)} \right)} \quad (11)$$

$$\lambda_o(t) = \frac{x(t)}{1+x(t)} C(t) + \dot{x}_{ref} \quad (12)$$

In addition sliding mode control has a *signum* part. So (11), (12) can be rewritten as follows:

$$C_p(t) = \frac{-\lambda_p + \dot{x}_{ref}(t)}{\left(-\frac{x(t)}{1+x(t)} \right)} + k_1 \text{sign}(s(t)) \quad (13)$$

$$\lambda_o(t) = \frac{x(t)}{1+x(t)} C(t) + \dot{x}_{ref} + k_2 \text{sign}(s(t)) \quad (14)$$

k_1 and k_2 for both systems are determined using trial and error. To reach a chattering free action, a *tangent hyperbolic* is used instead *signum* function. So, the control input for both systems are as follows:

$$C_p(t) = \frac{-\lambda_p + \dot{x}_{ref}(t)}{\left(-\frac{x(t)}{1+x(t)} \right)} + 100000 \tanh(s(t)) \quad (15)$$

$$\lambda_o(t) = \frac{x(t)}{1+x(t)} C(t) + \dot{x}_{ref} - 100000 \tanh(s(t)) \quad (16)$$

The simulation results are depicted in Figs. 3, 4 and 5 for premium traffic, and in Figs. 6, 7 and 8 for ordinary traffic. Figs. 3 and 6 show $x(t)$ with $x_{ref}(t)$ for Premium and Ordinary classes, respectively while good behavior for rising and settling of $x(t)$ is evident. The input and output rates of Premium buffer are shown in Figs. 4 and 5, respectively. Figs. 7 and 8 shows the input and output rates for the Ordinary buffers as well. To investigate the robustness of proposed controller, the round trip time delay and uncertainty is applied to the system as follows:

$$G(x(t)) = \left(1 + \frac{10}{100} \right) \frac{x(t)}{1+x(t)} \quad (17)$$

$$\tau = 3 \text{ m sec}$$

Figs. 9 and 10 shows the set point tracking behavior of $x_o(t)$ and $x_p(t)$, respectively with above conditions. It is evident that the performance of $x_p(t)$ with the proposed control action does not vary much; so the above uncertainty does not effect on the closed-loop system very much. The performance of $x_o(t)$ is a little worst than the case of without delay. It means that our proposed robust controller still needs to be improved to compensate the effect of round trip time delay.

5 Conclusion

In this paper, sliding mode Controller was applied to congestion control in Differentiated-Services networks. The growing demand of computer usage requires efficient ways of managing network traffic in order to avoid or at least limit the level of congestion in cases where increases in bandwidth are not desirable or possible. In this paper, a differentiated-services network framework was assumed and the control strategy was formulated for three types of services: Premium Service, Ordinary Service, and Best Effort Service. The proposed sliding mode control action demonstrated robust performance against round trip time delay and uncertainty. Some computer simulations showed good and satisfactory performance for the proposed controller.

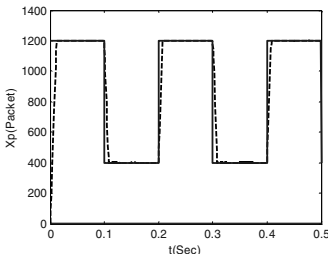


Fig. 3. $x_p^{\text{ref}}(t)$ and $x_p(t)$.

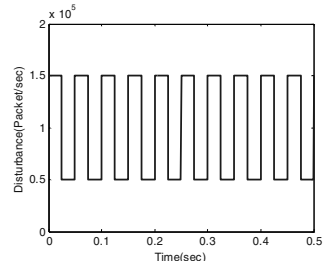


Fig. 4. Input rate of Premium's buffer.

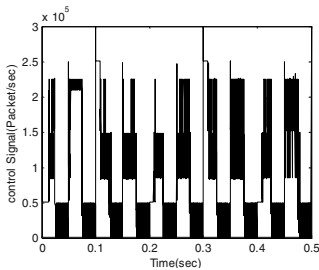


Fig. 5. Output rate of Premium's buffer.

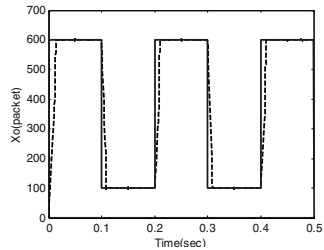


Fig. 6. $x_o^{\text{ref}}(t)$ and $x_o(t)$.

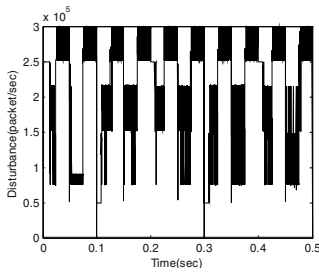


Fig. 7. Input rate of Ordinary's buffer.

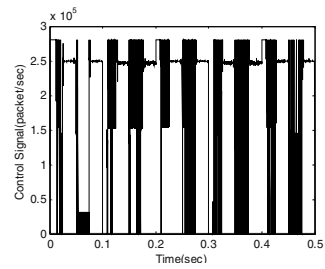


Fig. 8. Output rate of Premium's buffer

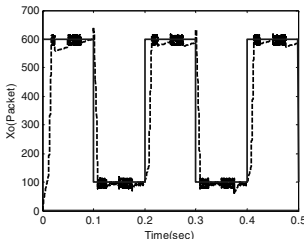


Fig. 9. $x_p^{\text{ref}}(t)$ and $x_p(t)$ with uncertainty and delay.

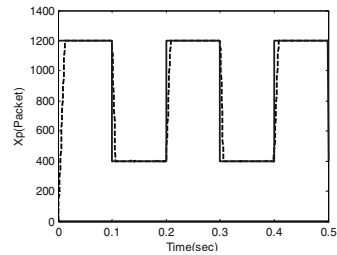


Fig. 10. $x_p^{\text{ref}}(t)$ and $x_p(t)$ with uncertainty and delay.

References

- [1] Rohrs C.E., Berry R.A. and O'Halek S.J., A Control Engineer's Look at ATM Congestion Avoidance, *IEEE GLOBECOM'95*, Singapore, 1995.
- [2] Keshav S., A control theoretic approach to flow control, *ACM SIGCOMM'91*, Zurich, Switzerland, 1991.
- [3] Benmohamed L. and Yang Y.T., A Control-Theoretic ABR Explicit Rate Algorithm for ATM Switches with Per-VC Queuing, *Infocom 98*, 1998.
- [4] Kolarov and Ramamurthy G., A control theoretic approach to the design of an explicit rate controller for ABR service, *IEEE/ACM Transactions on Networking*, October 1999.
- [5] Pitsillides A., Ioannou P. and Tipper D., Integrated control of connection admission, flow rate, and bandwidth for ATM based networks, *IEEE INFOCOM'96*, 15th Conference on Computer Communications, San Francisco, USA, March 1996, pp. 785–793.
- [6] Pitsillides and Lambert J., Adaptive congestion control in ATM based networks: quality of service with high utilization, *Journal of Computer Communications*, 20, 1997, pp. 1239–1258.
- [7] Barbashin E. A. and Geraschenko E. I. On speeding up sliding modes in automatic control systems. *Differentsialniye Uravneniya*, Vol. 1, 25–32, 1965.
- [8] Utkin, V. I. Variable structure systems with sliding mode. *IEEE Transactions on Automatic Control*, 22(2), April, 1977.
- [9] Utkin, V. I. Sliding modes in control optimization. New York, *Springer- Verlag*, 1992.

- [10] Ebrahimirad, H., Jalili-Kharaajoo, M., Yazdanpanah, M.J. and Labibi, B., Feedback linearization with sliding mode control of current and arc length of GMAW systems, *4th IFAC Sym. Robust Control Design, ROCOND2003*, Italy, June 2003.
- [11] Ebrahimirad, H., Vaez-Zadeh, S., and Jalili-Kharaajoo, M., Speed Control of PM Synchronous Motor: Comparison of Sliding Mode and PI Controllers, *IEEE CCA*, Istanbul, Turkey, June, 2003, pp 99–102.
- [12] Ebrahimirad, H., Vaez-Zadeh, S., and Jalili-Kharaajoo, M., Rubust Sliding Mode Control Applied to Speed Control of PM Synchronous Motors, *IEEE SCS2003*, Iasi, Romania, July, 2003, pp 317–320.
- [13] Stepanenko, Y. & Su, C. Variable structure control of robot manipulator with nonlinear sliding manifolds. *INT. J. Control*, 58(2), 285–300, 1993.
- [14] Xu, J.-X. & Cao, W.-J. Synthesized sliding mode control of a single-like flexible robot. *Int. J. control* 73(3), 197–209, 2000.
- [15] Bartolini, G. & Pydynowski, P. An improved, chattering free, VSC: scheme for uncertain dynamics systems. *IEEE Transaction on Automatic Control*, 41(8), August, 1996.
- [16] Bartolini, G., Ferrara, A. & Usai, E. Chattering avoidance by second-order sliding mode control. *IEEE Transactions on Automatic Control*, 43(2), February, 1998.
- [17] Blake, S., et al. An architecture for Differentiated Services. *Request for Comments RFC 2475, Internet Engineering Task Force*, 1998.
- [18] Nichols, K., et al. Definition of the differentiated Services Field in the Ipv4 and Ipv6 Headers. *Request for Comments RFC 2474, Internet Engineering Task Force*, 1998.
- [19] Jacobson, V., Nichols, K., & Poduri, K. An Expedited Forwarding PHB. *Request for Comments RFC 2598, Internet Engineering Task Force*, 1999.
- [20] Heinamen, J., Baker, F., Weiss, W., & Wroclawski, J. Assured forwarding PHB Group. *Request for Comments RFC 2597, Internet Engineering Task Force*, 1999.
- [21] Clark, D., & Fang, W. Explicit allocation of best effort packet delivery service. *IEEE/ACM Transactions on Networking*, 6(4), pp.362–373, 1998.
- [22] Sharma, S., D. Tipper, Approximate models for the study of nonstationary queues and their applications to communication networks, *IEEE ICC 93*, May 1993.
- [23] Shakkottai, S., R. Srikant, How Good are Deterministic Fluid Models of Internet Congestion Control, *IEEE*, June 2002
- [24] Tipper D., Sandareshan M. K., Numerical Methods for modeling Computer Networks Under Non-stationary Conditions, *IEEE Journal of Selected Areas in Communications*, Dec. 1990.
- [25] Filipiak J., Modelling and Control of Dynamic Flows in Communication Networks, *Springer-Verlag*, 1988.
- [26] Rossides L., Pitsillides A. and Ioannou P., Non-linear Congestion control: Comparison of a fluid flow based model with OPNET simulated ATM switch model, *TR-99-1*, Dept. Computer Science, University of Cyprus, 1999.
- [27] Pitsillides A. and Ioannou P., Non-linear Controllers for Congestion Control in Differentiated Services Networks, *TR-99-1*, Dept. Computer Science, University of Cyprus, 2001.

Application of Robust Fuzzy Adaptive Second-Order Sliding-Mode Control to Active Queue Management

Mahdi Jalili-Kharaajoo

Young Researchers Club
Islamic Azad University, Tehran, Iran
P.O. Box: 14395/1355, Tehran, Iran
mahdijalili@ece.ut.ac.ir

Abstract. Active Queue Management (AQM) takes a trade-off between link utilization and delay experienced by data packets. From control point of view, it is rational to regard AQM as a typical regulation system. In this paper, a new fuzzy adaptive second order sliding mode controller is designed for the objective of AQM. In the proposed method, the sliding parameter is adapted using fuzzy logic. Some computer simulations are provided to show the effectiveness of the proposed control action. Simulation results show the fuzzy logic based adaptive second order sliding mode has better performance in comparison with classic second order sliding mode control action in providing efficient queue management.

1 Introduction

TCP congestion control mechanism, while necessary and powerful, are not sufficient to provide good service in all circumstances, specially with the rapid growth in size and the strong requirements to Quality of Service (QoS) support, because there is a limit to how much control can be accomplished at end system. It is needed to implement some measures in the intermediate nodes to complement the end system congestion avoidance mechanisms. Active Queue Management (AQM), as one class of packet dropping/marketing mechanism in the router queue, has been recently proposed to support the end-to-end congestion control in the Internet [1]. It has been a very active research area in the Internet community. The goals of AQM are (1) reduce the average length of queue in routers and thereby decrease the end-to-end delay experimented by packets, and (2) ensure the network resources to be used efficiently by reducing the packet loss that occurs when queues overflow. AQM highlights the tradeoff between delay and throughput. By keeping the average queue size small, AQM will have the ability to provide greater capacity to accommodate nature-occurring burst without dropping packets, at the same time, reduce the delays seen by flow, this is very particularly important for real-time interactive applications. RED [2] was originally proposed to achieve fairness among sources with different burst attributes and to control queue length, which just meets the requirements of AQM. However, many subsequent studies verified that RED is unstable and too sensitive to parameter configuration, and tuning of RED has been proved to be a difficult job [3,4].

During the last two decades, Variable Structure Control (VSC) and Sliding Mode Control (SMC) have gained significant interest and are gradually accepted by practicing control engineers [5-8]. There are two main advantages of this approach. Firstly, the dynamic behavior of the system may be tailored by the particular choice of switching functions. Secondly, the closed-loop response becomes totally insensitive to a particular class of uncertainty. In addition, the ability to specify performance directly makes sliding mode control attractive from the design perspective [9]. A phenomenon that usually occurs in SMC is the problem of chattering that can be greatly reduced using higher order sliding controllers [10,11].

Fuzzy logic controllers have been developed and applied to nonlinear system for the last two decades. The most attractive feature of fuzzy logic control is that the expert knowledge can be easily incorporated into the control laws. Although a fuzzy logic controller is similar to a SMC [12], the combination of fuzzy logic control and sliding mode control still is of research interest due to the fact that the stability of a general fuzzy logic controller is difficult to prove whereas the stability of a sliding mode controller is inherent. In recent years many applications of fuzzy sliding mode control have been introduced [13]. The majority of research effort of combining fuzzy logic control and sliding mode control has been spent on how to use fuzzy logic to approximate the control command as a nonlinear function of sliding surface within the boundary layer [12,13,14].

The intuition and heuristic design is not always scientific and reasonable under any conditions. Of course, since Internet is a rather complex huge system, it is very difficult to have a full-scale and systematic comprehension, but importance has been considerably noted. The mathematical modeling of the Internet is the first step to have an in-depth understanding, and the algorithms designed based on the rational model should be more reliable than one original from intuition. In some of the references, the nonlinear dynamic model for TCP flow control has been utilized and some controllers like PI and Adaptive Virtual Queue Algorithm have been designed for that [15-19]. Although PI controller successfully related some limitations of RED, for instance, the queue length and dropping/mark probability are decoupled, whenever the queue length can be easily controlled to the desired value; the system has relatively high stability margin. The shortcomings of PI controller are also obvious. The modification of probability excessively depends on buffer size. As a result, for small buffer the system exhibits sluggishness. Secondly, for small reference queue length, the system tends to performance poorly, which is unfavorable to achieve the goal of AQM because small queue length implies small queue waiting delay. Thirdly, the status of actual network is rapidly changeable, so we believe that it is problematic and unrealistic, at least inaccurate, to take the network as a linear and constant system just like the designing of PI controller. Affirmatively, the algorithm based on this assumption should have limited validity, such as inability against disturbance or noise. We need more robust controller to adapt complex and mutable network environment, which will be our motivation and aim in this study. In the research, we will apply one of the advanced robust control theory, variable structure second order sliding mode control, to design the AQM controller. The sliding parameter of second order sliding mode controller is adapted using fuzzy logic [14], which results in better and more robust response in comparison with classic second order sliding mode control action.

2 TCP Flow Control Model

In [15], a nonlinear dynamic model for TCP flow control has been developed based on fluid-flow theory. This model can be stated as follows

$$\begin{cases} \frac{dW(t)}{dt} = \frac{1}{R(t)} - \frac{W(t)W(t-R(t))}{2R(t)} p(t-R(t)) \\ \frac{dq(t)}{dt} = \frac{N(t)}{R(t)} W(t) - C(t) \end{cases} \quad (1)$$

The above nonlinear and time-varying system was approximated as a linear constant system by small-signal linearization about an operating point [20] (Fig. 1). In the block diagram, $C(s)$ and $G(s)$ are the controller and the plant, respectively. The meaning of parameters presented in Fig. 1 are as following

$$\begin{aligned} K(t) &= \frac{[R(t)C(t)]^3}{[2N(t)]^2}, \quad T_1(t) = R(t), \\ T_2(t) &= \frac{R^2(t)C(t)}{2N(t)} \end{aligned} \quad (2)$$

where

$C(t)$: Link capacity (packets/sec)

q_o : Queue reference value

$N(t)$: Load factor, i.e., number of active sessions

$R(t)$: Round-trip time (RTT), $R(t) = 2(q(t)/C(t) + T_p)$, T_p is the fixed propagation delay

$p(t)$: Dropping/marking probability

$q(t)$: Instantaneous queue

We believe that the AQM controller designed with the simplified and inaccurate linear constant model should not be optimal, because the actual network is very changeful; the state parameters are hardly kept at a constant value for a long time. Moreover, the equations (1) only take consideration into the fast retransmission and fast recovery, but ignore the timeout mechanism caused by lacking of enough duplicated ACK, which is very usual in burst and short-lived services. In addition to, there are many non-respective UDP flows besides TCP connections in networks; they are also not included in equations (1). These mismatches in model will have negative impact on the performance of controller designed with the approach depending with the accurate model. For the changeable network, the robust control should be an appropriate choice to design controller for AQM. The variable structure sliding mode control action is one of the best that can help us.

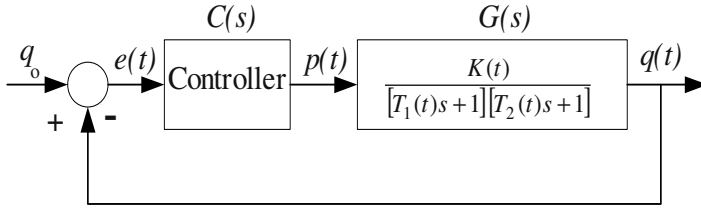


Fig. 1. Block diagram of AQM control system.

3 Fuzzy Adaptive Second-Order Sliding-Mode Controller Design

The accompanying (sometimes dangerous) vibrations are termed “chattering”. The higher the order of an output variable derivative where the high frequency discontinuity first appears, the less visible the vibrations of the variable itself will be. Thus, the remedy to avoid chattering is to move the switching to the higher order derivatives of the control signal. The problem is how to preserve the main feature of sliding modes: exact maintenance of constraints under conditions of uncertainty. Such sliding modes were discovered and termed “higher order sliding modes” (HOSM) [9,10,11]. These sliding modes may attract trajectories in finite time like the standard ones or may be asymptotically stable. Being moved to the higher derivatives of the control, the switching is no longer dangerous, since it takes place within the inner circuits of the control system (mostly in a computer) and not within the actuator. HOSM may provide for up-to-its-order precision with respect to the measurement time step, as compared to the standard (first order) sliding mode whose precision is proportional to the measurement step.

Consider a nonlinear system

$$\dot{x}(t) = f(t, x) + g(t, x)u(t) \quad (1)$$

where $f(t, x)$ and $g(t, x)$ are smooth uncertain functions and $u(t)$ is the control command. In the design of sliding mode, sliding surface is defined as a function of state variables

$$s = s(x, t). \quad (2)$$

Consider local coordinates $y_1 = s$ and $y_2 = \dot{s}$, after a proper initialization phase, the second order sliding mode control problem is equivalent to the finite time stabilization problem for the uncertain second order system

$$\begin{cases} \dot{y}_1 = y_2 \\ \dot{y}_2 = \varphi(y_1, y_2, x_2, u, t) + \gamma(y_1, y_2, t)v(t) \end{cases} \quad (3)$$

where $v(t) = \dot{u}(t)$.

In the above equations $y_2(t)$ is generally unknown, but $\varphi(t, x)$ and $\gamma(t, x)$ can be bounded as

$$|\varphi(t, x)| \leq \Phi, \quad 0 < \Gamma_m < \gamma(t, x) < \Gamma_M, \quad \Phi > 0. \quad (4)$$

Being historically the first known second order sliding controller, that algorithm features twisting around the origin of second order sliding phase plane $y_1 - y_2$. The trajectories perform an infinite number of rotations while converging in finite time. The vibration magnitudes along the axes as well as the rotation times decrease in geometric progression. The control derivative value commutes at each axis crossing, which requires availability of the sign of the sliding-variable time-derivative y_2 .

The control algorithm is defined by the following control law, in which the condition on $|u|$ provides for $|u| \leq 1$

$$\dot{u}(t) = \begin{cases} -u & \text{if } |u| > 1 \\ -V_m \text{sign}(y_1) & \text{if } y_1 y_2 \leq 1, |u| \leq 1 \\ -V_M \text{sign}(y_1) & \text{if } y_1 y_2 > 1, |u| \leq 1 \end{cases} \quad (5)$$

The corresponding sufficient conditions for the finite-time convergence to the sliding surface are

$$V_M > V_m, V_M > \frac{\Gamma_M V_m + 2\Phi}{\Gamma_m}, V_m > \frac{\Phi}{\Gamma_m}, V_m > \frac{4\Gamma_M}{s_0} \quad (6)$$

where $s_0 > |s|$.

Fuzzy logic is mainly introduced to provide tools for dealing with uncertainty [21]. We cannot determine the change of C with respect to the system action precisely. Thus, we will use a fuzzy logic controller to supervise the sliding controller. In order to obtain fuzzy rules of updating the slope of the sliding curve $s = Cx_1 + x_2$, the approximate form of the phase plane around the sliding surface is depicted in Fig. 2. For the operating points A to H we can define the slope change as presented in Table 1. The fuzzy rules are presented in Table 2 and the corresponding membership functions are shown in Fig. 3.

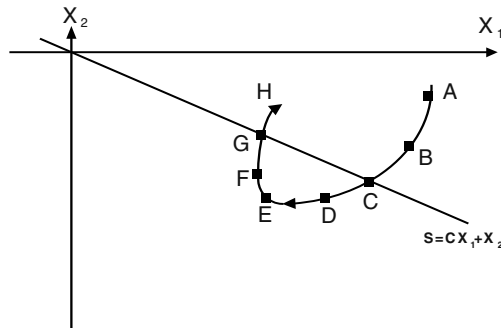


Fig. 2. Phase trajectory around sliding surface.

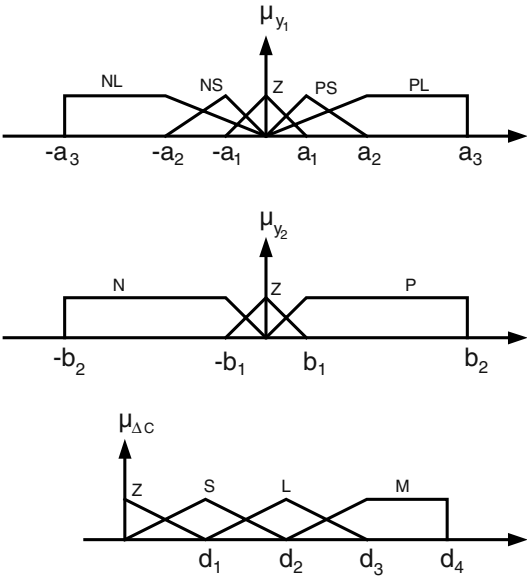


Fig. 3. Fuzzy membership functions.

Table 1. Proper change in C for different operating points.

ΔC	$y_2(t)$	$y_1(t)$	Operating point
Zero	Negative	Positive and large	A
Small	Negative	Positive and small	B
Medium	Negative	Zero	C
Large	Negative	Negative and small	D
Medium	Zero	Negative and large	E
Small	Positive	Negative and small	F
Zero	Positive	Zero	G
Zero	Positive	Positive and small, large	H

Table 2. Fuzzy rules.

		$y_1(t)$					
$y_2(t)$		NL	NS	Z	Z	PL	
	P	M	S	Z	Z	Z	
	Z	M	M	M	Z	Z	
	N	M	L	M	S	Z	

4 Simulation Results

The network topology used for simulation, is depicted in Fig. 4. The only bottleneck link lies between node A and node B. the buffer size of node A is 300 packets, and default size of the packet is 500 bytes. All sources are classed into three groups. The first one includes N_1 greedy sustained FTP application sources, the second one is composed of N_2 burst HTTP connections, each connection has 10 sessions, and the number of pages per session is 3. The thirds one has N_3 UDP sources, which follow the exponential service model, the idle and burst time are 10000msec and 1000msec, respectively, and the sending rate during "on" duration is 40kbps. We introduced short-lived HTTP flows and non-responsive UDP services into the router in order to generate a more realistic scenario, because it is very important for a perfect AQM scheme to achieve full bandwidth utilization in the presence of noise and disturbance introduced by these flows. The links between node A and all sources have the same capacity and propagation delay pair (L_1, τ_1) . The pair (L_2, τ_2) and (L_3, τ_3) define the parameter of links AB and BC, respectively.

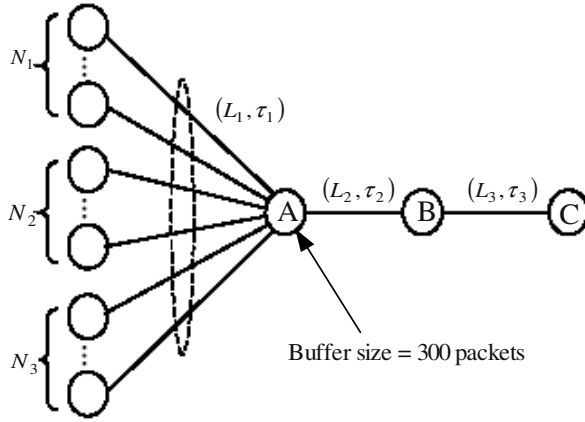


Fig. 4. The simulation network topology.

In the first study, we will use the most general network configuration to testify whether the Fuzzy Adaptive Second-Order Sliding-Mode Controller (FASOSMC) can reach the goals of AQM, and freely control the queue length to stabilize at the arbitrary expected value. Therefore, given that $(L_1, \tau_1) = (10Mbps, 15ms)$, $(L_2, \tau_2) = (15Mbps, 15ms)$, $(L_3, \tau_3) = (45Mbps, 15ms)$. $N_1 = 270$, $N_2 = N_3 = 0$. Let the expected queue length equal to 75 packets. The initial value of C is set to 5. The fuzzy controller values are chosen as follows:

$$a_1=1 \quad a_2=5 \quad a_3=400 \quad b_1=0.5 \quad b_2=400 \quad d_1=0.05 \quad d_2=0.2 \quad d_3=0.5 \quad d_4=2$$

The instantaneous queue length using the proposed FASOSMC is depicted in Fig. 5. After a very short regulating process, the queue settles down its stable operating point. RED algorithm is unable to accurately control the queue length to the desired value. The queue length varies with network loads. The load is heavier the queue length is longer. Attempting to control queue length through decreasing the interval between high and low thresholds, then it is likely to lead queue oscillation. Although Second-Order Sliding-Mode controller (SOSMC) could regulate the queue to the fixed point, the integrated performance needs to be improved, such as the transient process is too long and the fluctuation in steady state is great, for small queue length, which lows the link utilization. The queue evaluation of router A, controlled by SOSMC controller ($q_o=75$ packets), is plotted in Fig. 6. Evidently, SOSMC controller takes the longer time to settle down the reference point.

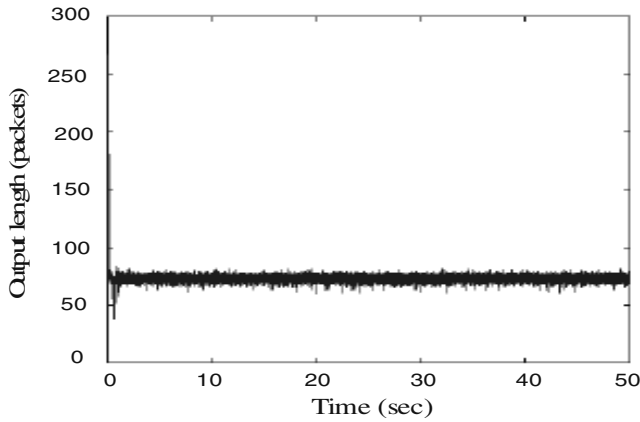


Fig. 5. Queue evaluation (FASOSMC).

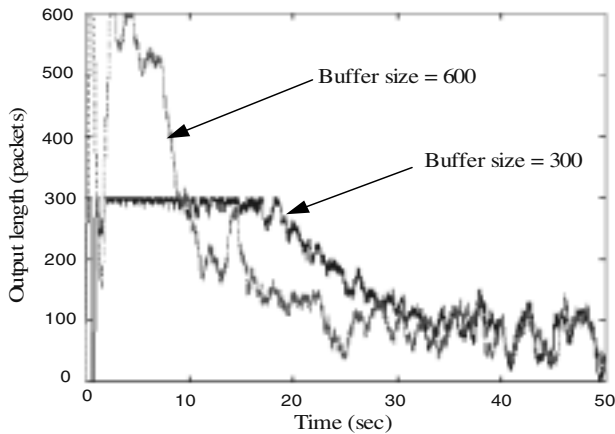


Fig. 6. Queue evaluation (SOSMC).

Considering the requirement of the steady state performance, it is impractical to increase the difference between a and b to speed up the response of SOSMC. With the higher sampling frequency, the computation will be significantly exhausted. The only feasible way is to add the buffer size. In order to illustrate this ability, we redo the above simulation with 600 packets buffer size, which the results are also plotted in Fig. 6. Indeed, the large buffer is able to enhance the responsibility of SOSMC, but this ability is limited, moreover it seems to be wasteful. Conversely, the FASOSMC has the ideal performance without any additional regulation mechanism. In order to evaluate the performance in steady state, we calculate the average and the standard deviation of the queue length in steady state. For the convenience of comparison, choose the queue length between 40 and 50 seconds as sample data. In this case, the standard deviation of SOSMC (32.3336) is much larger than that of FASOSMC (2.5928). Fig. 7 presents the case of small reference queue length. Except $q_o = 15$, the other parameters are unchangeable.

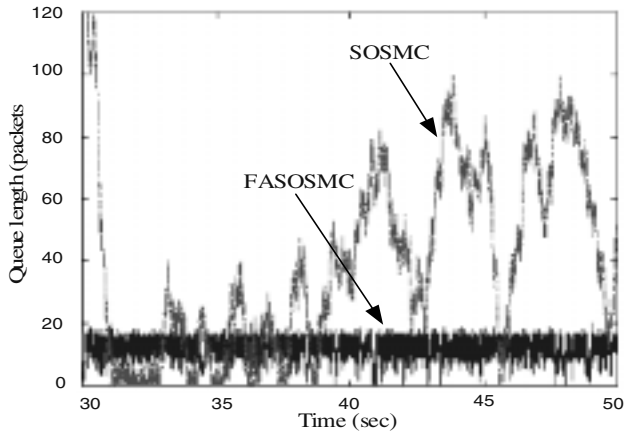


Fig. 7. Small expected queue ($q_o=15$).

In this section, Firstly, let $N_1 = 270, N_2 = 400, N_3 = 0$, the evaluation of queue size is shown in Fig. 8. As it can be seen, the proposed FASOSMC has better performance than that of SOSMC. Next, given that $N_1 = 270, N_2 = 0, N_3 = 50$, we further investigate performance against the disturbance caused by the non-responsive UDP flows. Fig. 9 shows the results, obviously, SOSMC is very sensitive to this disturbance, while FASOSMC operates in a relatively stable state. The queue fluctuation increases with introducing the UDP flows, but the variance is too much smaller comparing with SOSMC.

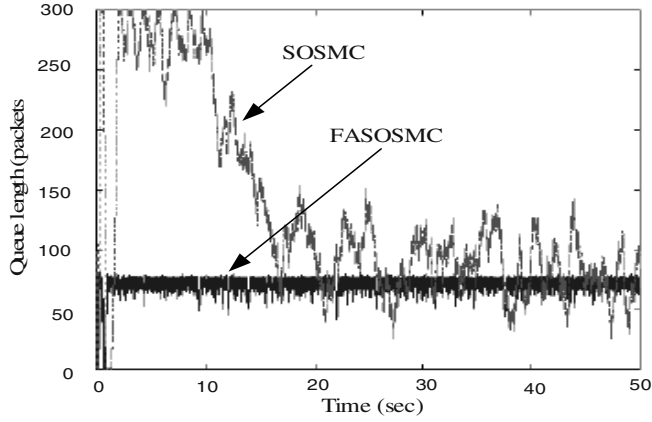


Fig. 8. Queue evaluation (FTP+HTTP).

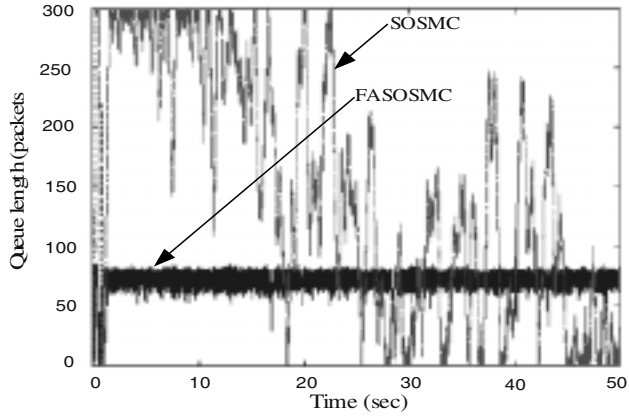


Fig. 9. Queue evaluation (FTP+UDP).

5 Conclusions

In this paper, a fuzzy adaptive second-order sliding-mode controller was designed for the objective of active queue management. For this purpose, a linearized model of the TCP flow was considered. We applied FASOSMC to this system because this advanced robust control methodology is insensitive to system dynamic parameters and is capable of being against disturbance and noise, which is very suitable for the mutable network environment. We took a complete comparison between performance of the proposed FASOSMC and classical SOSMC under various scenarios. The conclusion was that the integrated performance of FASOSMC was superior to that of SOSMC. FASOSMC was very responsive, stable and robust, especially for the small reference queue system, but its performance was inferior when active TCP sessions were relatively small. Thus, it will be very imperious to design the controller suitable for light load, and then integrate it with FASOSMC using classical adaptive control technology.

References

- [1] Barden, B. et al., Recommendation on queue management and congestion avoidance in the internet, REC2309, April 1998.
- [2] Floyd, S. and Jacobson, V., Random early detection gateway for congestion avoidance, IEEE/ACM Trans. Networking, August 1993.
- [3] Firoiu, V. and Borden, M., A study of active queue management for congestion control, in Proc. INFOCOM, March 2000.
- [4] May, M., Bonald, T. and Bolot, T., Analytic evaluation of RED performance, in Proc. INFOCOM, March 2000.
- [5] Utkin, V.I., Sliding Modes in Control and Optimization, Springer Verlag(1992).
- [6] Edwards, C. and S.K. Spurgeon, Sliding mode control . Taylor & Francis Ltd(1998).
- [7] Jalili-Kharaajoo, M., Mesgharpour Tousi, M., Bagherzadeh, H. and Esna Ashari, A., Sliding mode control of voltage-controlled magnetic levitation systems, IEEE CCA, Istanbul, Turkey, June, 2003, pp. 83–86.
- [8] Jalili-Kharaajoo, M. and Feizi, S., Sliding mode control applied to motion control of linear structural systems under earthquake excitation, IFAC workshop, DECOM-TT 2003, Istanbul, Turkey, June, 2003.
- [9] Young, D. K., V.I. Utkin and O. Özgüner, “A control engineer’s guide to sliding mode control”, IEEE Transaction on Control System Technology, vol. 7, No. 3, pp.621–629(1999).
- [10] Bartolini, G., A. Ferrara, E. Punta and E. Usai, “Application of a second order sliding mode control to constrained manipulators”, EUCA, IFAC and IEEE European Control Conference, Bruxelles, Belgium(1997).
- [11] Sira-Ramirez, H., “Dynamic second order sliding mode control of the hovercraft vessel”., IEEE Transaction on Control System Technology, vol. 10, No. 6, pp.880–865(2002).
- [12] Kung, C.C. and S.C. Lin, “A Fuzzy-Sliding Mode Controller Design”, IEEE International Conference on Systems Engineering, pp.608–611(1992).
- [13] Boverie, S., P. Cerf and J.M. Le Quellec, “Fuzzy Sliding Mode Control Application to Idle Speed Control”, IEEE International Conference on Fuzzy Systems, pp.974–977(1994).
- [14] Jalili-Kharaajoo, M., and Ebrahimirad, H., Improvement of second order sliding mode control applied to position control of induction motors using fuzzy logic, Lecture Notes in Artificial Intelligence (2715), Proc. IFSA2003, Istanbul, Turkey, 2003.
- [15] Misra, V., Gong, W.B. and Towsley, D., Fluid-based analysis of network of AQM routers supporting TCP flows with an application to RED, in Proc. ACM/SIGCOMM, 2000.
- [16] Holot, C., Misra, V., Towsley, D. and Gong, W.B., On designing improved controllers for AQM routers supporting TCP flows, in Proc. INFOCOM, 2001.
- [17] Misra, V., Gong, W.B. and Towsley, D., Analysis and design an adaptive virtual queue (AVQ) algorithm for active queue management, in Proc. ACM/SIGCOMM, 2001.
- [18] Kelly, F.P., Maulloo, A. and Tan, D., Rate control in communication networks, Journal of the Operation research Society, 49, pp.237–252, 1998.
- [19] Athuraliya, S., Lapsley, D.E. and Low, S.H., Random early marking for internet congestion control, in Proc. Globecom, 1999.
- [20] Holot, C., Misra, V., Towsley, D. and Gong, W.B., A control theoretic analysis of RED, in Proc. INFOCOM, 2001.
- [21] Zadeh, L.A., Fuzzy sets, *Inf. Control* (1965), 338–353.

Graceful Degradation of Transport Layer in Mobile Internet

Yosuke Matsushita, Takahiro Matsuda, and Miki Yamamoto

Graduate School of Engineering, Osaka University,
2-1, Yamadaoka, Suita, Osaka 565-0871, Japan

yosuke@post.comm.eng.osaka-u.ac.jp <http://www2b.comm.eng.osaka-u.ac.jp>

Abstract. In mobile Internet, a handover may result in significant degradation of transmission performance. Especially in a shared medium wireless network, handover traffic which moves into a new wireless access network competes with existing traffic for available bandwidth of shared medium wireless channel. This interaction between handover traffic and existing traffic causes significant performance degradation of both of traffic. In this paper, we propose a new concept of *Transport Layer Graceful Degradation*. We think handover traffic should increase its available bandwidth gradually and available bandwidth of existing traffic is to be decreased slowly to bandwidth which achieves fair share of shared medium with handover traffic. In the paper, we propose a new bandwidth control method for TCP traffic which achieves our proposed concept, transport layer graceful degradation. Performance evaluation results show that our proposed scheme can improve TCP throughput performance of both handover traffic and existing traffic.

1 Introduction

In mobile networks, the movement of a mobile node causes handover where the mobile node changes its access point to another access point. Layer 3 protocols such as Mobile IP[1] maintain layer 3 connectivity even when a route between source and destination nodes has changed due to a handover. However, even if connectivity is kept by these network layer protocols, packet losses during handover cannot be resolved because packets of a handover flow have a possibility of being miss-routed to an old access point until the handover processing of a layer 3 protocol is completed. When TCP is used in transport layer, these bursty loss caused by handover will severely degrade throughput performance. Therefore, many researchers have studied techniques to improve TCP throughput performance during handover[2]-[7]. The aim of these researches is to improve throughput performance handover TCP session.

When a shared medium wireless channel, e.g. IEEE 802.11[11], is used in a wireless access network, interaction between handover traffic and existing traffic occurs. When TCP performance improvement is applied for handover TCP, handover TCP may inject large volume of packets in a new access network. In this case, these injected traffic of handover TCP may cause congestion or bandwidth

competition in a wireless access network. Thus, not only handover TCP session but also existing TCP sessions will suffer from bursty packet losses due to congestion or bandwidth competition. This means both of handover TCP session and existing TCP sessions severely degrades their throughput performance.

In the paper, we propose a new concept, transport layer graceful degradation. This concept is handover session and existing sessions should gracefully degrade its performance to fair share bandwidth. Under our transport layer graceful degradation concept, handover session should be gradually increase its available bandwidth and existing sessions should be degraded gracefully.

In the paper, we apply this transport layer graceful degradation concept to TCP session and propose a new bandwidth control method for handover TCP session. In our proposed method, we assume that the network has a hierarchical structure[8]- [10] which is divided into a core network and sub-domains. The bandwidth competition occurs in these sub-domains. Based on the widely-used hierarchical model of Internet, this is a natural assumption since core network has abundant capacity compared to the sub-domain, while the sub-domains process handover much faster than the core. Under this assumption, the proposed bandwidth control is implemented within a sub-domain router. The hierarchical network structure is suited to the network supported control. A handover processing within a sub-domain is operated at several gateway routers in the sub-domain which are referred as Mobility Anchor Points(MAPs) in Hierarchical Mobile IPv6 (HMIPv6)[8]. Therefore, it is possible to implement our proposed schemes in these gateway routers alone, because all packets flowing into a sub-domain have to pass through these gateway routers and then, gets exposed our proposed control mechanism.

The rest of the paper is organized as follows. In section II, the effect of handover and the concept of Graceful Degradation is introduced. In section III, we explain the architecture of mobile Internet and the network support technique in the mobile Internet. In section IV, we propose the new bandwidth control for Graceful Degradation. In section V and VI, we evaluate our proposed scheme with computer simulation. In section VII, we present our conclusions.

2 Graceful Degradation

2.1 The Effect of Handover

Mobile networks are usually designed in a cellular fashion, where each cell covers a particular area. Users inside this area are served from an access point (AP). When a mobile node (MN) is moving from one cell to another, a procedure named handover must be followed. Various layers are related to handover. Physical and data link layers are related to switching connectivity between an MN and an AP. If the network layer is based on IP, mobility support protocol such as Mobile IP[1] is necessary because the current IP assumes that a node's IP address uniquely identifies the node's point of attachment to the Internet.

When TCP is used in the transport layer, it is known that handover flows suffer significant throughput degradation due to bursty packet losses during han-

dovers. The cause of these losses is that the packets are miss-routed to the previous access point and discarded until the handover process under the network layer is completed. Because TCP is designed on the assumption that all packet losses occur due to congestion, TCP sender activates its congestion control techniques unnecessary for these packets and suffers the significant throughput degradation as a result. Many schemes are proposed to compensate for this performance degradation[2]-[7].

Here, we examine the effect of handovers from the viewpoints of handover flows. A mobile node can monitor the signal strengths in the wireless antennas and can detect impending handovers. Therefore, handover flow can avoid the bursty packet losses resulting from handovers using several techniques. One of these techniques is called bicast[7], in which the router transmits the same segment simultaneously to a current access point and candidate next access points while mobile node is moving. The mobile node avoids the bursty packet losses by receiving the packets from both access points. When this scheme is used, the handover induced packet loss is completely eliminated and the handover is concealed from the user.

Next, we consider the effect of handovers from the viewpoint of non-handover flows¹. Non-handover flows are not capable of detecting the impending handover of other flows, which make it difficult to implement any preventive technique. As a result, when non-handover flows share a bottleneck link with handover flow, the impact of the handover flow is even stronger when it is coupled with the bicast scheme. The bicast scheme avoids handover related degradation for handover flow, which in turn results in injection of a heavier traffic burst when the handover flow is rerouted. This heavier traffic burst degrades performance of the non-handover flows.

When the access point or router can manage the bandwidth by some sort of QoS control such as circuit switch networks or RSVP (Resource reSerVation Protocol), it is possible to minimize the influence of the handover. The access point can block the handover flow, when it becomes clear in advance that if the access point accepts the subscription of handover flow the required performance cannot be maintained. However, in the random access network like wireless LANs etc., it is difficult to avoid the bandwidth competition. When bandwidth competition occurs, the many packets arrive to the shared link leads to significant congestion. This indicate that the bandwidth competition problem of handover and non-handover flows is an important technical subject.

2.2 The Concept of Graceful Degradation

Almost researches published thus far try to conceal the influence of handover from users. As far as we know, no research has considered yet the influence of handover flows on non-handover flows.

¹ In this paper, we define non-handover flows as existing flows on the link where handover flow is injected

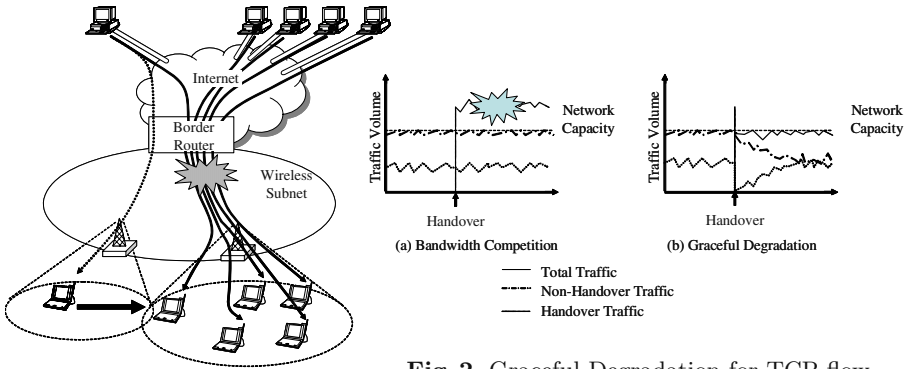


Fig. 2. Graceful Degradation for TCP flow

Fig. 1. The Effect of Handover

In general, rate-regulated flows such as TCP flows converge to a steady state when sharing a bottleneck link. The handover flow creates a transient period when it enters a bottleneck link where each non-handover flow fairly shared it beforehand. In this transient period, the handover flow may inject large volume traffic and lead to unfair condition. At the end of this transient period, however, both the non-handover and handover flows will fairly share the link capacity.

Graceful Degradation is defined as a technique which makes both flows transit gracefully between two steady states before and after handovers. “Graceful” refers to a gradual transition without significant performance degradation. The purpose of Graceful Degradation is not to prevent the significant performance degradation effect of the bandwidth competition².

It is inevitable that the handover flow degrades the throughput of the non-handover flows. When a handover flow enters the link which has been shared by n number of flows, the fair share of each flow decreases to $1/(n+1)$ from $1/n$ regardless of whether Graceful Degradation is applied or not.

3 The Traffic Control Technique by Network Support

In this paper, we propose a network supported traffic control which realizes the Graceful Degradation. In the following, we show that the structure of the mobile networks is suitable for our proposed network support technology.

3.1 The Network Architecture of Mobile Internet

Mobile IP was proposed to solve inherent issues in IP routing protocol where packets destined to a mobile node would not be able to reach it while the mobile node is away from its home link. Mobile IP provides information about a mobile node’s current location with a care-of-address. All packets addressed to the

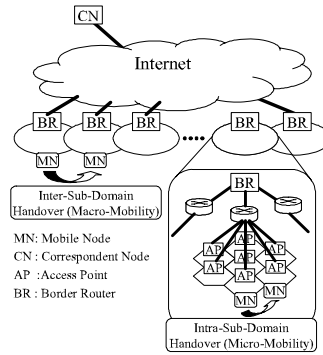


Fig. 3. Network Structure of Hierarchical Mobile Internet

mobile node's home address are transparently routed to its care-of-address. Mobile IP describes how a mobile node can change its point of subnet attachment from the access router at one access point to the other, i.e., handover. However, Mobile IP lacks support for fast handover control. This handover latency, which is the time required to re-establish a care-of-address on the new subnet, is unacceptably long to support real-time or delay sensitive traffic. Therefore, a number of IP micro-mobility protocols, which complement the Mobile IP by providing fast handover control, have been proposed. Generally, these micro-mobility protocols use a hierarchical network structure as shown in Fig.3. In the network structure, the whole network is divided into sub-domains. The micro-mobility protocols differentiate the intra-sub-domain mobility (micro-mobility) from the inter-sub-domain mobility (macro-mobility). Namely, when a mobile node moves within a sub-domain, the micro-mobility protocol hides the mobile node's mobility from the node the mobile node is communicating with. The mobile node's location information such as care-of-address is locally updated by signaling only within its subnet, while in the existing Mobile IP, the care-of-address is updated by signaling between the mobile node and the Home Agent (HA) in the mobile node's home link.

3.2 Traffic Control in Mobile Internet

In the hierarchical mobile Internet, handover initiated route changes within a sub-domain are handled in the corresponding border routers. Figure 4 shows that all the traffic that flows into a sub-domain crosses through the sub-domain's border router. Therefore, services built on network support technology can be efficiently offered to many users by mounting the network support function into these border routers. The hierarchical mobile Internet topology thus can support the deployment of network supporting technologies. In our proposed system, we control the traffic of a sub network by adding a bandwidth control function to the border router of the subnet. This is because that the border router can detect

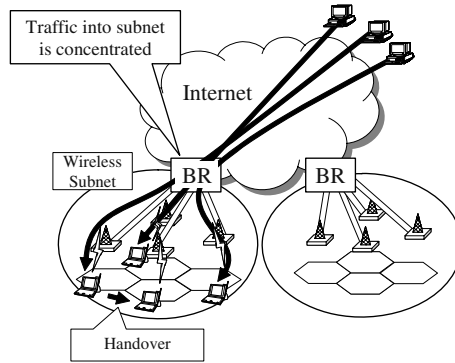


Fig. 4. Traffic Control in Mobile Network

the handover flow from the information of the binding update which is used to notify the new care-of-address of the mobile node after the handover.

4 Bandwidth Control for Graceful Degradation of TCP Traffic

In this section, we give an overview of our proposed scheme. In our proposed scheme, when the route of the handover flow changes, the border router limits the inbound traffic of the handover flow. The border router caches the packets of the handover flow temporarily and sends these packets to the output buffer little by little. The method which limits the traffic works as follows: For each handover flow, the border router initializes a window mechanism. The size of the window wnd determines the number of buffered handover packets that can be released into the bottleneck. The router increases the size of the window gradually. The operation of the proposed algorithm in detail:

- (1) When the handover flow is rerouted, the border router sets up a timer, which signals the end of the operation of our proposed scheme (Control Time τ).
- (2) wnd is set to 1, and a cached packet of the handover flow is released.
- (3) each returning ACK from the mobile node increases the size of the window with 1, that is $wnd = wnd + 1$.
- (4) When the sequence number of ACK does not come in order (i.e. a duplicate ACK is received), the router assumes that packet loss was occurred due to bandwidth competition and the system returns to (2).
- (5) If any ACKs do not come within time δ , the algorithm assumes that packet loss was occurred, and the system returns to (2). δ is calculated by the same principle as the retransmission timer of TCP[12] using round trip delay between a border router and a mobile receiver.
- (6) When the timer expires, the cache is released.

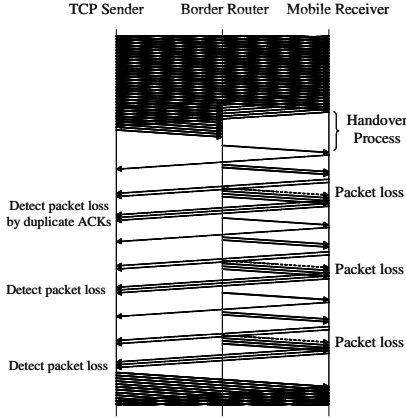


Fig. 5. Process of Our Scheme

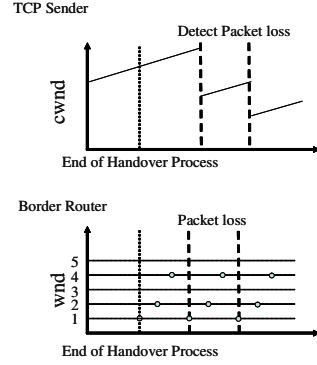


Fig. 6. Window Size in Proposed Scheme

From (2) and (3), it is clear that our proposed scheme is similar to the slow start algorithm of TCP. However, the packets of the handover flow are queued in the cache. Also, this algorithm is fundamentally different from slow start, since slow start is invoked as a result of the burst loss due to the bandwidth competition. To detect the burst losses, the TCP sender using slow start algorithm has to wait for the retransmission timer to expire. In this case, the throughput of both the handover flow and the non-handover flow degrades significantly. Moreover, because the slow start is an end-to-end control, the time needed for the flows to reach their optimal rate depend on their RTT. On the other hand, because our algorithm is operated at the border router (near the MN), the response to update the *wnd* is faster and there is no need to wait retransmission timeout either. In our proposed scheme, the transmission rate of the sender is adjusted only by a few packet losses (not bursty) during the control time (Fig.5 and Fig.6).

5 Simulation Model

In this section, we evaluate the performance of our proposed scheme with computer simulation. In order to the understand fundamental effect of the bandwidth competition problem between the handover and non-handover flows, we employ a simple network model. In our network model, we assume that the mobile node moves within a sub-domain, i.e., no macro-mobility is taken into account. We further assume that packet loss due to bit error or collision doesn't occur in the wireless link.

Figure 7 shows the simulation model . There are 6 TCP flows, where all nodes belong to the same access point. 5 of these nodes are fixed receivers, and one of them is a mobile receiver. The non-handover flows which have fixed receivers share a bottleneck link. The handover flow which has a mobile receiver

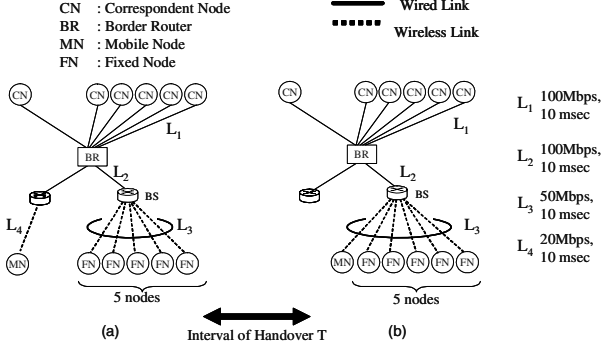


Fig. 7. Simulation Model

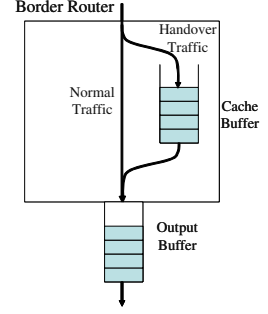


Fig. 8. Buffer in Border Router

gets rerouted to the bottleneck link of the non-handover flows due to a handover (Fig.7(a) and Fig.7(b)). The mobile node has an exponential distribution handover interval with an average of T . The TCP sources have endless supply of data to send. The time necessary for the route change at the data link and network layers is 60[msec]. We use TCP-Reno as the transport layer protocol which is one of the most popular TCP version.

The other parameters are as follows. B is the output buffer size of the IP router, and B_c is the cache buffer size of the handover flow. When our proposed scheme is operating, the packets of the handover flow are cached in the cache buffer and transmitted to the output one in response to the size of wnd (Fig.8).

- Packet (IP datagram) Length 1500 bytes
- $B = 300[packets]$, $B_c = \infty$

6 Simulation Result

6.1 Evaluation Index

In this paper, we evaluate the transitional (Fig.9-Fig.11) and the average characteristic (Fig.12-Fig.17). The transitional characteristics are presented by the throughput transition when the MN moves from the state on Fig.7(a) to the state on Fig.7(b). The average characteristics are presented by average throughput when the MN moves between the state Fig.7(a) and the state Fig.7(b) alternately. We express the average throughput of the handover flow as γ_M and the average throughput of the non-handover flow as γ_F . γ_M and γ_F are measured in Fig.7(b). Namely, they represent throughput during the bandwidth competition. $\tilde{\gamma}$ marks the average throughput when all nodes including the handover flow are fixed Fig.7(b). We evaluate the proposed scheme using the throughput ratios $\gamma_M/\tilde{\gamma}$, $\gamma_F/\tilde{\gamma}$ in the state Fig.7(b). These throughput ratios show the fairness between the handover flow and non-handover flows when handover occurs. That is, if the throughput ratio is close to 1, it can be said that the handover

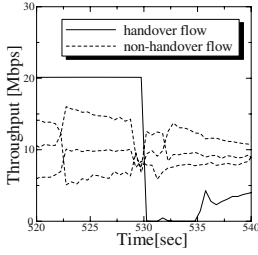


Fig. 9. Throughput Transition without Proposed Scheme

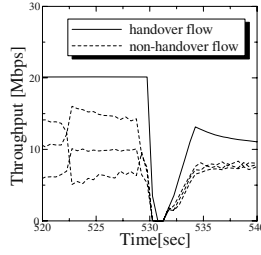


Fig. 10. Throughput Transition with Bicast Scheme

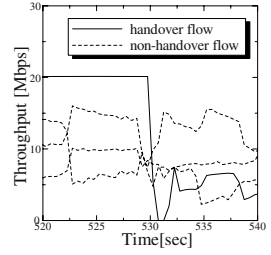


Fig. 11. Throughput Transition with Proposed Scheme

flow and non-handover flows share the battle neck bandwidth fairly, and graceful degradation is realized.

We compare our proposed scheme against the w/o proposed scheme and the bicast scheme, where w/o proposed scheme is the scheme that does not prevent the packet loss due to handover and bandwidth competition problem. The effects of the bandwidth competition in this case can be ignored the throughput of handover flow is degraded due to the retransmission timeout and slow start techniques of the TCP. The handover induced packet loss however can not be ignored. Therefore comparing our proposed scheme to the against the w/o proposed scheme reveals the effects of preventive technique for packet losses of handover flow. In the bicast scheme, however, the packet losses due to handover are perfectly avoided. Therefore, comparing our proposed scheme with the bi-casting scheme means evaluating the effect of Graceful Degradation without the handover induced packet losses.

Our proposed system is designed assuming that handover induced packet losses are avoided by some other applied technique. In the proposed scheme, the packets are buffered at the border router during the handover process to avoid miss-forwarding. Although our proposed system can work together with many other schemes aimed at eliminating handover induced packet losses, the reason we use the buffering scheme is that this scheme does not waste the bandwidth. Since our proposed system operate at the congested networks, it is not a good idea to use our proposed scheme together with schemes which waste the network bandwidth to improve the handover of flows.

6.2 Transition of Throughput

Fig.9 - Fig.11 show the transition of throughput when the handover occurs at time 530 seconds. Although there are 5 non-handover flows in our computer simulation, we present the throughput of only 3 of them. Figure 9 shows the transition of throughput using the w/o proposed scheme. In this figure, the throughput of handover flow shows significant degradation when the handover occurs. This is due to bursty losses during handover. Moreover, the throughput of the handover flow plummets to 0 for a few seconds. The introduction of the

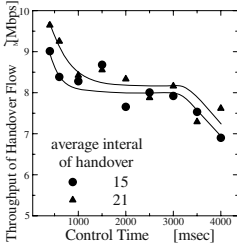


Fig. 12. Throughput Performance of Handover Flow vs. Control Time τ

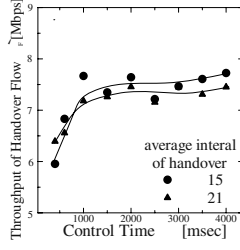


Fig. 13. Throughput Performance of Non-Handover Flow vs. Control Time τ

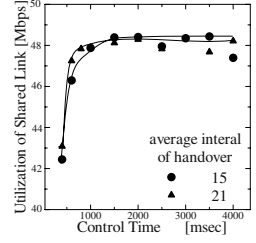


Fig. 14. Utilization of Shared Link vs. Control Time τ

handover into the new link creates congestion in the link, and as a result of this congestion the retransmitted packet of the handover flow also gets lost, which forces the TCP of the handover flow into exponential back off. On the other hand, when the bicast scheme is used on Fig.10, both handover and non-handover flows show significant degradation. This is due to the bursty losses which is the result of the bandwidth competition problem. These bursty losses force both handover and non-handover flows into the slow-start phase of the TCP restarting communication. Based on these results, it can be said that the bicast scheme has a bad influence on the non-handover flows and that it also lowers the utilization of the link.

Figure 11 shows the transition of throughput using our proposed scheme, which shows improved performance compared the w/o case. In our scheme, bursty losses are avoided by buffering and using the slow start algorithm when introducing the handover flow into the link. As of the non-handover flows, bursty losses are avoided by controlling the amount of incoming packets. As a result, the handover flow increases its throughput gradually, while the non-handover flow decreases its throughput gracefully.

6.3 Throughput Performance vs. the Control Time τ

On Fig.12 and Fig.13, we evaluate the throughput performance γ_M and γ_F versus the control time τ . γ_M is decreasing, τ is larger than 3000[msec]. When τ is the larger than 3000[msec], the proposed scheme is not effective. This is because as follows. If the packet loss occurs between the border router and the mobile node during control time, wnd is reset to 1 as explained in (4)(5). Therefore, although the handover flow and non-handover flow can share the bandwidth of the link without throughput degradation by decreasing the packet in cache buffer, the throughput of the handover flow may be suppressed by the proposed scheme. In this case, the γ_M becomes lower as the control time becomes longer.

When τ is small, γ_F is low due to the bandwidth competition problem. This is caused by the packets in the cache buffer. In our proposed scheme time τ after the handover, the border router releases the cache and the packets in the

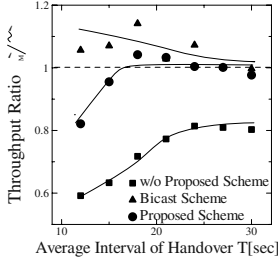


Fig. 15. Throughput Performance of Handover Flow vs. T

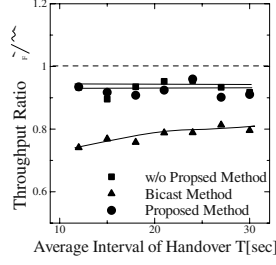


Fig. 16. Throughput Performance of Non-Handover Flow vs. T

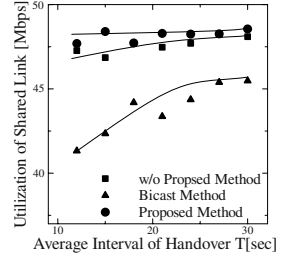


Fig. 17. Utilization of Shared Link vs. T

cache buffer is transmitted to the output buffer. Therefore, when many packets remain in the cache buffer, the bandwidth competition problem occurs after all. Then the throughput of non-handover flow are recovered as control time passes. From the above results, we can see that the throughput performance of the handover flow degrades when the τ is too large and the throughput performance of non-handover flow degrades when the τ is too small. Figure 14 shows the utilization of the shared link versus τ . The link utilization is close to 1 when the τ is larger than 2000[msec]. Therefore, in the following, we show the simulation results when the τ is 2000[msec] from the view point the link utilization is kept high and the throughput of the handover flow is not degraded.

6.4 Throughput Performance vs. Handover Interval T

Figures 15 and 16 show the throughput ratio of the handover flow and the non-handover flow versus the average handover interval T . Figure 15 shows that the $\gamma_M/\tilde{\gamma}$ of the w/o proposed scheme is much lower than 1, and in the case of the bicast scheme the $\gamma_M/\tilde{\gamma}$ is more than 1. In the w/o proposed scheme, retransmission timeout and slow start of the sender occurs and exponential back off may even occur. Therefore, the throughput of the handover flow can't increase smoothly after the handover. In the bicast scheme, the bandwidth competition problem occurs because the packets are bicasted two routes before and after the handover. Because the mobile node receives the packets from both base stations, the mobile node receives the packets from the non-congested access point even if the packet loss occurs on the congested link. That is, the handover flow occupies the shared bandwidth as the throughput of the non-handover flows degrade. On the other hand, in the proposed scheme, both $\gamma_M/\tilde{\gamma}$ and $\gamma_F/\tilde{\gamma}$ are close to 1.

Figure 17 shows the utilization of the shared link. Our proposed scheme has higher throughput performance than other schemes proving that our scheme is effective in realizing the Graceful Degradation. The reason that the throughput performance of our proposed scheme is higher than one of w/o proposed scheme is that the packets of handover flow tend to bursty because the congestion threshold is high and the synchronization problem of TCP occurs.

7 Conclusion

In this paper, we have proposed a system which realizes Graceful Degradation of TCP traffic by bandwidth control in the border router in the hierarchical mobile Internet. We have showed that our proposed scheme can avoid the bad influence of handover flows and non-handover flows by using computer simulation. Graceful Degradation which is examined in this paper is a new concept in mobile Internet. It is expected that this idea demonstrates the most greatest effect in a heterogeneous environment where at the point user density or access network etc.. As a nest step in our future work, we plan to clarify the relation between the performance of our proposed scheme and the bandwidth delay product.

References

1. D. B. Hohnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," Internet draft, draft-ietf-mobile-ipv6-18, work in progress, June 2002.
2. T. Goff, J. Morosaki, and D. S. Phatak, "Freeze-TCP: A True End-to-End TCP Enhancement Mechanism for Mobile Environments," Proc. of IEEE INFOCOM 2000, pp. 1537–1545, 2000
3. K. Brown and S. Singh. "M-TCP: TCP for Mobile Cellular Networks," ACM Computer Communication Review, vol. 27, No.5, pp.19–43, Oct. 1997.
4. P. R. Calhoun, T. Hiller, J. Kempf, P.J.MaCann, C. Pairla, S. Thalanaly, and A. Singh, "Foreign Agent Assisted Hand-off," Internet draft, November 2000.
5. A. E. Yegin, C. E. Perkins, G. Domety, K.El-Malki, and M. Khalil, "Fast Handovers for Mobile IPv6," Internet draft, March 2002.
6. G. Krishnamurthi, R. C. Chalmers, and C. E. Perkins, "Buffer Management for Smooth Handovers in Mobile IPv6," Internet draft, July 2000.
7. A.O'Neill, S.Corson, and G.Tsirtsis, " Generalized IP Handoff," Internet draft, August2002.
8. H. Soliman, C. Castelluccia, K.El-Malki, and L. Bellier, "Hierarchical MIPv6 mobility management (HMIPv6) Internet draft," July 2002.
9. R. Ramjee, T. La Porta, S. thuel, K. Varadhan, and S. Y. Wang, wireless networks, "HAWAII: a domain-based approach for supporting mobility in wide-area wireless network" Proc. of ICNP '99 pp. 283–292, 1999.
10. A.G.Valko, "Cellular IP – A New Approach to Internet Host Mobility," ACM Computer Communication Review, pp. 50–65, January 1999.
11. B. P. Croe et al. , "IEEE 802.11 Wireless Local Area Networks," IEEE Communications Magazine, Vol. 35, No. 9, pp. 116–126, September 1997.
12. W. R. Stevens, TCP/IP Illustrated Vol. 1, The Protocols, Addison Westley, 1994.

Implementing Ad Hoc to Terrestrial Network Gateways

Jonathan McGee, Manish Karir, and John S. Baras

Center for Satellite and Hybrid Communication Networks
Department of Electrical and Computer Engineering
University of Maryland, College Park, MD 20742, USA
{mcgee,karir,baras}@isr.umd.edu

Abstract. In this paper we describe our experience of implementing a gateway between ad hoc and terrestrial routing protocols. Our implementation of the gateway includes support for both a unicast routing protocol as well as a multicast routing protocol. Though we limit our implementation to a particular set of protocols, we believe that the principles involved can easily be applied to other routing protocols. In particular, in this paper we detail our work on implementing a gateway between a network running MOSPF on a wired terrestrial network interface and MAODV on a wireless ad hoc network interface. Although we focus primarily on the single gateway scenario, we also discuss complications that arise from the use of multiple gateways and illustrate the potential failures that can arise in those scenarios.

1 Introduction

While there has been a lot of work on ad hoc routing protocols, the operation of a gateway between ad hoc and terrestrial domains has received relatively little attention. There is an increasing interest in this problem as people are gradually realizing that ad hoc networks probably will not operate as stand alone networks. There will be the need at some point to connect back to a terrestrial network.

There are various problems associated with building efficient routing protocol gateways between ad hoc and terrestrial networks. The core of the problem arises from the very different characteristics of these networks. Wired terrestrial networks are considered relatively stable in topology, therefore, routing protocols that have evolved to run in those environments are largely based on a proactive approach of maintaining routing information. Ad hoc networks, on the other hand, are more suited to use reactive approaches although proactive approaches to routing in ad hoc networks also exist. Reactive approaches are best suited for ad hoc networks because of the expected mobility of the nodes within these networks. A proactive approach would simply generate too much control overhead with even moderate mobility and network size. Hybrid approaches that attempt to optimize these two contrasting approaches have also been developed for ad hoc networks.

The problem of developing a gateway between a *proactive* terrestrial routing protocol and a *proactive* ad hoc network routing protocol is relatively simple as the routing protocols are similar in nature. Each protocol aims at constructing a detailed routing table that reflects the connectivity of the entire network and the gateway is then simply

responsible for *exporting* and *importing* routing information from each protocol. This problem is similar in nature to the exchange of routing information between OSPF and BGP in terrestrial networks, for example.

The exchange of information between a *proactive* terrestrial routing protocol and a *reactive* ad hoc network routing protocol is a bit more challenging as the type of information that a proactive routing protocol requires is not the same as the information that a reactive ad hoc routing protocol makes available. Therefore, to achieve interoperability between the two requires implementing either complex registration/deregistration for ad hoc network nodes or developing a proxy routing daemon that can provide abstracted information on behalf of the ad hoc network to the terrestrial routing protocol.

In this paper we take the second approach where the gateway provides abstracted ad hoc network information to the terrestrial routing protocol. We address both the cases of unicast as well as multicast routing protocols. To demonstrate our approach we implemented our prototype solution using MOSPF as the terrestrial routing protocol and MAODV as the ad hoc network routing protocol. Our approach does have the drawback that details of node location and state within the ad hoc network are lost. This can create problems related to efficiency when multiple gateways are present. We discuss these issues at the end of this paper.

The rest of this paper is organized as follows: In section 2 we discuss some related work. Section 3 presents a description of the problem for both unicast and multicast routing. Section 4 describes our implementation of an ad hoc to terrestrial network gateway. Section 5 provides a discussion regarding how our work can be extended to the scenario of multiple gateways. Finally section 6 describes our conclusions and some directions for future work.

2 Related Work

There has been some work recently in addressing the problem of interfacing wireless ad hoc networks with terrestrial networks. Most of the work has focused on providing Internet connectivity to the ad hoc network. In particular, people have studied in some detail the use of mobileIP and registration based protocols to solve the problems of integrating the two networks.

[1] addresses the issue of providing global Internet access for MANETs, in particular, focusing on routing, the problem of global address resolution and gateway discovery. They provide an excellent discussion of the issues and an architecture for attempting to solve these problems. Similarly, [2] presents a scheme for providing Internet connectivity for ad hoc mobile networks. They provide a mechanism to enable cooperation between MobileIP and the AODV routing protocol. They use simulation results to validate their architecture by showing that it can maintain high throughput while keeping the overall control overhead low. [3] provides a brief description of how AODV can be used for providing inter-networking between wireless ad hoc networks and the IPv6 Internet. The primary focus is to describe the process of gateway discovery by using a multicast group, and the determination of a node address.

[4] differs from the above works in that it describes an actual software implementation of an integrated connectivity solution. In architecture, the solution described in [4] is

similar to the solutions of [1] and [2], in that it combines the use of AODV in the wireless ad hoc network and MobileIP in the terrestrial network. They describe the implementation of gateway nodes that run both AODV and MobileIP software and are responsible for providing the connectivity.

While all the works described so far have focused on the use of AODV, [5] on the other hand proposes the use of a protocol independent gateway for ad hoc networks. This cluster gateway is responsible for providing Internet connectivity for the ad hoc network by acting as both a service access point as well as a MobileIP foreign agent. Ad hoc nodes register themselves with the cluster gateway, which makes its location known by periodic advertisements.

Although [6] uses ODMRP which has multicast routing capability, they focus on the unicast performance of ODMRP in an extended hybrid network consisting of both ad hoc and terrestrial nodes. They present an implementation of an extension of ODMRP which allows the ad hoc network to dynamically connect to the wired network. They do not, however, address how multicast operation can be achieved in the same scenario.

In [7] the authors examine in detail the use of MobileIP for providing Internet connectivity for Ad Hoc networks. They consider both proactive as well as reactive approaches to maintaining registration with the MobileIP foreign agent and present simulation results to show the benefits of their hybrid approach. They also do not consider multicast operation.

The solutions presented so far often involve complex protocols to achieve their design objectives. To the best of our knowledge, a unified gateway that supports both unicast as well as multicast routing between a terrestrial and an ad hoc routing protocol has not been implemented before.

3 Problem Description

The problem of providing connectivity between ad hoc and terrestrial routing protocols is becoming increasingly important. This connectivity problem can be divided into two parts: the first dealing with addressing unicast routing connectivity and the second addressing interoperation between multicast routing protocols in the two routing domains.

Numerous proposals have attempted to solve the first problem as described in the section 2, however the proposed solutions do not possess certain key characteristics. We attempted to base our solution on the following design principles:

- The solution should not add unnecessary complexity to an already complex problem.
- The solution should attempt to be as close in principle as possible to the way the problem of inter-domain routing is handled in the current Internet.
- The solution should be implementable in a realistic scenario.
- The solution should require only minimal or no modifications to terrestrial routing protocols. The terrestrial routing protocols are widely accepted standards, whereas the ad hoc routing protocols are still being developed.
- The solution should include support for both unicast as well as multicast routing protocols.

In the following subsections we further characterize the two problem areas: of unicast and multicast routing gateways. The basic differences in the way these are implemented in the Internet motivates us to consider these as two separate problems. We later combine them into the same gateway implementation.

3.1 Unicast Routing

In AODV the establishment of a route is made on demand. This is in direct contrast to most proactive terrestrial routing protocols that assemble all possible routes as they learn of other hosts. In order to provide a *native* integration between ad hoc and terrestrial networks, it would be necessary for the gateway to determine the availability of individual hosts on a periodic basis so they can be advertised to the terrestrial domain.

The above problem is composed of two distinct parts. The first is how nodes from the ad hoc domain reach nodes in the terrestrial domain and the second is how nodes in the terrestrial domain reach nodes in the ad hoc network domain. Solving the first requires that nodes in the ad hoc domain be able to distinguish between destinations that are within the ad hoc domain and those that are outside. Attempting to solve the second problem requires that a mechanism exist for notifying the terrestrial domain routing protocol which nodes are present in the ad hoc domain.

Complex protocols and solution can be devised to solve each of these, however, we argue that if we adopt a single simplifying assumption the solutions become almost trivial. By representing the ad hoc network as a single aggregated address space we can easily solve the problem for routing traffic in and out it. Moreover, this approach also has the very attractive benefit of greatly simplifying the problems of address configuration and duplicate address resolution in the ad hoc network.

3.2 Multicast Routing

In multicast routing packets sent to a group are distributed to many hosts in multiple networks. In a terrestrial network a host subscribed to a multicast group need only notify its router of its interest in receiving traffic for that multicast group using IGMP. The routers construct a tree to distribute the packets based on which multicast routing protocol is being used. Therefore, IGMP is responsible for maintaining group membership information at the local network level while a multicast routing protocol such as MOSPF or DVMRP is responsible for the formation of the multicast forwarding tree by providing information to the routers in-between the source and destinations of the multicast traffic.

In ad hoc networks since each node is also a router IGMP is inappropriate because there is no real *local* network. Instead the operation of maintenance of group membership is entirely performed by the multicast routing protocol. This difference in philosophy in the two types of networks creates a problem when we try to implement a gateway. Terrestrial multicast routing protocols assume the presence of IGMP on all interfaces to notify them of group membership changes. Once again, our goal is seamless inter-operation between the two domains such that nodes in the ad hoc domain are able to join multicast groups originating in the terrestrial domain and nodes in the terrestrial network are able to join multicast groups originating in the ad hoc network domain.

4 Ad Hoc to Terrestrial Network Gateway Description

In this section we describe our implementation of a gateway between an ad hoc network and a terrestrial network. For our implementation we used OSPF as the terrestrial routing protocol and AODV as the ad hoc network routing protocol. We selected OSPF as a good candidate as at least one implementation which was publicly available included support for multicast operation as well. We used the OSPF implementation from www.ospf.org. As this version of OSPF includes support for multicast operation in this paper we use the terms OSPF and MOSPF interchangeably in this paper. For our ad hoc network routing protocol we used the AODV-UU version of AODV. We modified AODV-UU to include support for the multicast extensions to AODV. In this paper we use the terms AODV and MAODV interchangeably to refer to this implementation. Both our choices of routing protocols provided unicast and multicast routing support in one software distribution. This greatly simplifies the task of building a unified gateway. We constructed our gateway node to have a wired and a wireless interface. It runs both the MOSPF software as well as our version of MAODV routing protocol.

4.1 Unicast Routing Gateway

As described in an earlier section, the problem of implementing a unicast routing gateway involves being able to determine and distinguish between nodes that are within the ad hoc network and those that are outside. This can be accomplished via the use of complex registration and query protocols. However, if we make a single assumption that the ad hoc network can be represented as a single aggregated address space, we no longer need complex registration protocols.

To ensure interoperability between a terrestrial network and an ad hoc network, the ad hoc network must take care to ensure that its nodes select globally unique addresses. In the case where the terrestrial network is the Internet, the address space would need to be assigned via some body such as the IANA. Therefore, it is natural to assign a fixed prefix for the ad hoc network domain.

Using this assumption we can configure the gateway to advertise the ad hoc network as a prefix. This solves the problem of allowing terrestrial nodes to reach nodes within the ad hoc domain as packets destined for the ad hoc domain will automatically be routed to the gateway node which will be responsible for forwarding them into the ad hoc domain. In our implementation we configured OSPF to manage both the terrestrial and ad hoc network as OSPF areas. This isn't completely necessary: simply advertising the ad hoc network as an external route is just as functional. No changes needed to be made to the OSPF daemon.

For the ad hoc domain nodes to be able to reach nodes in the terrestrial domain is also trivial, as each node knows by simply looking at the destination whether that address is in the ad hoc or terrestrial domain. To reach external nodes, ad hoc nodes simply send a RREQ messages as usual. When these RREQ messages reach the gateway, it will reply with a RREP message if that destination is in the terrestrial network.

This method also has the advantage of simplicity. There is no need for the gateway router running OSPF to keep track of the dynamics of the ad hoc network. The gateway acts as shield by advertising a single aggregate route to the terrestrial domain. This protects the terrestrial network from the large numbers of updates that mobile ad hoc nodes might otherwise generate.

4.2 Multicast Routing Gateway

Our approach to implementing the gateway between MOSPF and MAODV is similar to the approach we use for unicast routing: we make the ad hoc domain appear like a single network segment. MOSPF expects to hear IGMP messages on the wireless interface in order to forward multicast traffic onto it. We create a proxy agent on the gateway that performs the job of translating the MAODV group join/leave messages into appropriate IGMP messages and injects these messages into the IP stack for MOSPF to receive.

Therefore, for the multicast gateway it is necessary to configure MOSPF to treat the ad hoc network as a full OSPF area. Since requests and joins will be coming in on the ad hoc interface, it is necessary for the MOSPF daemon to be listening on this interface to be aware of multicast activity in the ad hoc network. The MAODV daemon on the gateway is modified to join all multicast groups it hears about. This is important because the gateway has to be able to inform the terrestrial network about the presence of these groups.

An alternate solution would have been to modify MOSPF to listen to MAODV messages and process them as IGMP messages, but this would require extensive modifications to MOSPF. Our approach has the desired property that we do not need to modify MOSPF. The only changes we needed to make were to the MAODV code that ran on the gateway.

When a multicast group exists in the terrestrial network and a node in the ad hoc domain wishes to join it, the node sends out a RREQ for that group; when this reaches the gateway our modified MAODV code injects an IGMP *group-join* message into the IP stack such that the MOSPF daemon sees there is a node on the wireless interface network that wishes to subscribe to that group. MOSPF will then process this injected IGMP message and propagate this interest out into the terrestrial domain.

When a multicast group exists in the ad hoc network and a node in the terrestrial domain wishes to join this group, it will send out IGMP *group-join* messages which will propagate this interest through the terrestrial network until it reaches the gateway node. The gateway node, by virtue of being a part of the ad hoc domain, is already subscribed to this group and will simply start forwarding multicast traffic from the wireless ad hoc network interface onto the wired terrestrial interface.

4.3 Implementation Details and Testbed Description

We implemented our gateway on nodes that ran Red Hat Linux 7.3 with a Linux 2.4.19 kernel from www.kernel.org. OSPFD version 2.0 from ospf.org was used as the terrestrial routing daemon and AODV-UU version 0.6 was used as the AODV implementation (user.it.uu.se/~henrikl/aodv/). Multicast additions to AODV-UU made by our group were applied to the source. In addition, the code was further modified to support the gateway operation as described in the previous sections. No changes were made to the MOSPF software.

We only needed to make minor modifications to AODV-UU to support correct operation as a unicast gateway. AODV-UU has a basic gateway mode where the configured host will automatically offer a default route for destinations off the network. In the version of AODV-UU we used (version 0.6), this mode did not properly function and was revised. One of the major changes was a fix to make the AODV daemon check the kernel's route table and only return a reply if an appropriate route was found.

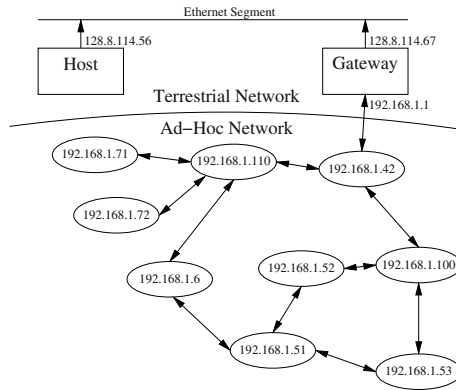


Fig. 1. Gateway Test Network

Modifying MAODV to support the multicast gateway capability, required significantly more work. Changes were made such that the gateway would join groups on reception of a group hello message and remove itself only once all upstream and downstream hosts have removed themselves. In addition, we needed to mimic the behavior of IGMP on the wireless interface of the gateway. A kernel module was created to inject packets into the IP stack. This module provides a directory in `/proc/net` with a node for each network device. A write to one of these nodes is converted into a single incoming packet. We then used this interface to inject IGMP packets into the IP stack.

Our test network topology is shown in Figure 1. We used appropriate subsets of this topology to test different functions of the gateway. We tested unicast connectivity in our testbed using a simple ping as well as `tcpdump` to analyze the traffic being received and forwarded on various segments to insure proper behavior. It was particularly important to ensure correct operation of the gateway when the ad hoc network contains nodes that are multiple hops away from the gateway. We were able to verify correct operation of the gateway in this scenario as well. Multicast connectivity was initially tested with a simple custom program to join a group and periodically transmit and receive multicast packets. In addition we also used *vic*, a common multicast video streaming application, to verify correct multicast connectivity. During these tests, both unicast and multicast route/forwarding tables were monitored to verify the proper behavior of the routing protocols.

5 Multiple Gateways Scenario

Until now we have focused our attention on the problem where the ad hoc and the terrestrial network have a single point of attachment. Next we turn to more complex scenarios where there are multiple points of attachment. In this section we illustrate some of the issues that arise when we take into consideration the possibility of multiple gateways and discuss some potential solutions to these problems.

5.1 Multiple Unicast/Multicast Gateways

There are several problems that arise in unicast routing protocol interoperability when we allow for scenarios where multiple gateways can exist. Some are related with with interaction between the nature of proactive and reactive unicast routing protocols, and some are particular to multicast routing. The problem with unicast routing are related to the inability of the gateways to obtain complete information regarding the availability and location of different nodes in the ad hoc network. This leads to inefficient routing and reachability problems when the network is partitioned. Multicast routing protocol inter-operation in the presence of multiple gateways is difficult because group membership management is even further complicated.

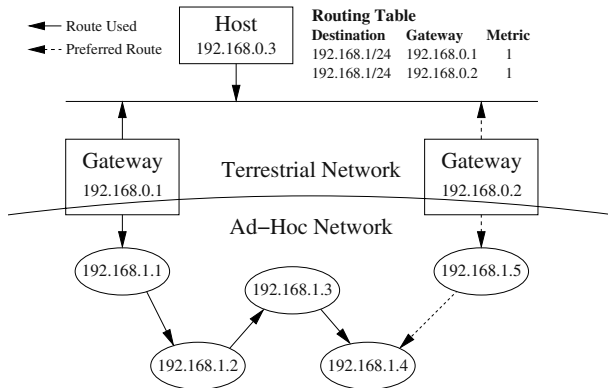


Fig. 2. Example of Poor Route Selection

1. *Inefficient routing paths:* This problem is illustrated in Figure 2. The ad hoc network can be reached via either gateway; however, the metrics that are advertised into the terrestrial network only reflect the reachability of the gateway in the terrestrial domain and do not reflect the number of hops inside the ad hoc network. This can result in inefficient paths being chosen from nodes in the terrestrial domain to the ad hoc domain. The decision of which gateway the terrestrial nodes choose is determined only by their distance to the gateway and not the total distance to the destination. It might have been possible to reach the destination node via fewer hops by choosing a gateway that was slightly further away in the terrestrial domain but closer to the final destination node in the ad hoc network. This problem does not exist for nodes in the ad hoc network attempting to reach nodes in the terrestrial domain, as the RREP messages that the gateways generate reflect the correct metrics of the node in the terrestrial network.
2. *Ad Hoc Network Partitions:* The gateway nodes only advertise the reachability of the ad hoc network, not that of individual nodes. When a partition occurs in the ad hoc network, all gateways will not be able to reach all nodes. This can result in a situation where traffic originating from the terrestrial network can get routed to a

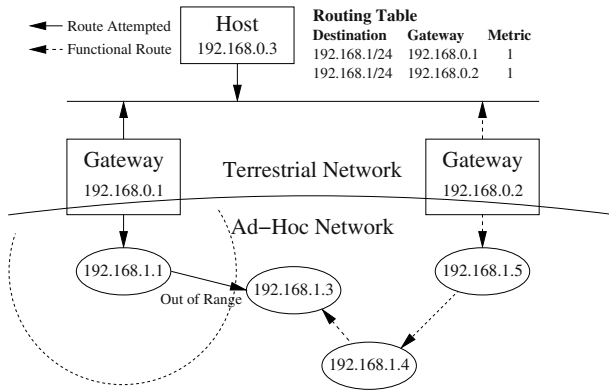


Fig. 3. Example of Unnecessary Failure due to Partition

gateway which is unable to reach a particular node even though that node might be reachable via another gateway. This problem is depicted in Figure 3.

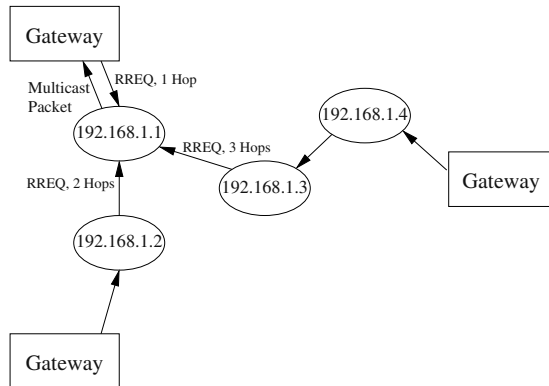


Fig. 4. Multicasting to a Non-existent Group

3. *Multicasting to a non-existent group from the ad hoc domain.* If a host wishes to message a group without joining the group using MAODV, it will send a RREQ without the *join* flag set. Several gateways will reply to this RREQ however, the host will pick only one path to the nearest gateway. This can create a problem as only the terrestrial network connected with that gateway will receive the muticast packets. This is shown in Figure 4.
4. *Inability to detect group prunes.* In order to conserve resources, it is desirable that gateways will prune groups that are no longer active. This can fail once multiple gateways have joined the group as shown in Figure 5. If there is only one gateway on the network, when the last interested member leaves the group, all intervening nodes between the gateway and that node will prune themselves. When multiple

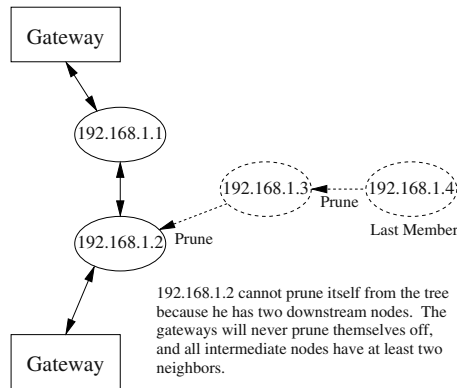


Fig. 5. Example of Inability to Prune Inactive Groups

gateways are in the group, atleast one node will have two active downstream nodes when the last member leaves, as shown in Figure 5. This node will be unable to prune itself and the group will persist despite having no active members.

5.2 Discussion

The problems described in the previous section related with unicast routing are a direct result of our attempts to insulate the terrestrial network from the frequent changes in the ad hoc network. There is no single solution for all these problems. One approach might be to use a registration based protocol to notify each gateway about nodes in the ad hoc network that it is responsible for. Another might be the use of a proactive protocol in the ad hoc domain, but this has its own drawbacks.

The problem with multicast routing are equally difficult. If one aggressively prunes idle groups in an attempt to recover resources, broadcasts from non-members to off-network groups may be lost. On the other hand, if one forms groups for every route request, many idle groups are created that will not be pruned. A modification to the multicast routing protocol could be added to allow gateways to test for the presence of completely inactive groups, but modifications to the protocol and all supporting software would be undesirable for many applications. Another solution would be a compromise between resource consumption and reliability of non-member multicasts. The gateways would have a soft-limit on number of active multicast groups. Once this limit is reached, idle groups begin to be pruned. As the number of groups approaches a hard-limit, the aggressiveness of the prune increases.

In summary, the use of multiple gateways between the same terrestrial and ad hoc networks can create scenarios where efficiency and even accuracy of the gateway is reduced. Therefore, we recommend that, care should be taken to ensure the proper functioning of the integrated network when multiple gateways are present, and scenarios such as the ones we have described should be avoided.

6 Conclusions and Future Work

In this paper we have presented an implementation of a gateway to provide interoperability between routing protocols in the ad hoc and terrestrial network domains. We have provided details of our design and our implementation. We hope that this will provide valuable information to other developers of routing gateways. While our solution works for this particular set of routing protocols, we are investigating how we can make our implementation more generic so that it is easy to implement the same functionality for any set of terrestrial and ad hoc routing protocols. In addition, we also discussed some issues related to realise internetworking in the presence of multiple terrestrial to ad hoc network gateways. We argue that due to the additional complexity introduced by multiple gateways, care should be taken to ensure the proper functioning of the integrated network in the presence of multiple gateways or only a single active gateway between the networks should be permitted.

We are currently attempting to validate our gateway approach on a network using a combination of RIP and DVMRP instead of MOSPF. In addition we are investigating ways of resolving the issues and improving the performance of our solution for scenarios where multiple gateways are present.

Acknowledgments. The material described in this paper is based upon work performed in collaboration with Telcordia Technologies and sponsored by the U.S. Army Research Lab. award number DAAD17-00-C-0115. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies either expressed or implied of the Army Research Lab or the U.S. Government.

References

1. B. Andreadis. Providing Internet Access to Mobile Ad Hoc Networks. *London Communications Symposium*, Sep 2002.
2. Y. Sun, E. M. Belding-Royer, and C. E. Perkins. Internet Connectivity for Ad Hoc Mobile Networks. *International Journal of Wireless Information Networks*, Apr 2002.
3. A. Nilsson, C. Perkins, R. Tuominen, A. J. Wakikawa, and Malinen J. T. AODV and IPv6 Internet Access for Ad Hoc Networks. *Mobile Computing and Communications Review*, Vol. 6, Number 3, Jul 2002.
4. C. Ahlund and A. Zaslavsky. Integration of Ad hoc Network and IP Network Capabilities for Mobile Hosts. *10th International Conference on Telecommunications (ICT)*, Feb 2003.
5. A. Striegel, R. Ramanujan, and J. Bonney. A Protocol Independent Internet Gateway for Ad-Hoc Wireless Networks. *Proceedings of Local Computer Networks (LCN) 2001*, Nov 2001.
6. S. H. Bae, S. Lee, and M. Gerla. Unicast Performance Analysis of Extended ODMRP in a Wired-to-Wireless Hybrid Ad Hoc Network. *IEEE International Conference on Communications and Networks (ICCN)*, Las Vegas, Oct 2000.
7. P. Ratanchandani and R. Kravets. A Hybrid Approach to Internet Connectivity for Mobile Ad Hoc Networks. *Proceedings of IEEE Wireless Communications and Networking Conference(WCNC)*, Mar 2003.

Connecting Wireless Sensornets with TCP/IP Networks

Adam Dunkels^{1,2}, Juan Alonso¹, Thiemo Voigt¹, Hartmut Ritter³, and Jochen Schiller³

¹ Swedish Institute of Computer Science, Box 1263, SE-164 29 Kista, Sweden
{adam,alonso,thiemo}@sics.se

² Department of Computer Science and Engineering, Mälardalen University, Box 883,
SE-721 23 Västerås, Sweden.

³ Institute of Computer Science, Freie Universität Berlin, Takustr. 9, D-14195 Berlin, Germany
{hritter,schiller}@inf.fu-berlin.de

Abstract. Wireless sensor networks are based on the collaborative efforts of many small wireless sensor nodes, which collectively are able to form networks through which sensor information can be gathered. Such networks usually cannot operate in complete isolation, but must be connected to an external network through which monitoring and controlling entities can reach the sensornet. As TCP/IP, the Internet protocol suite, has become the de-facto standard for large-scale networking, it is interesting to be able to connect sensornets to TCP/IP networks. In this paper, we discuss three different ways to connect sensor networks with TCP/IP networks: proxy architectures, DTN overlays, and TCP/IP for sensor networks. We conclude that the methods are in some senses orthogonal and that combinations are possible, but that TCP/IP for sensor networks currently has a number of issues that require further research before TCP/IP can be a viable protocol family for sensor networking.

1 Introduction

Wireless sensor networks is an information gathering paradigm based on the collective efforts of many small wireless sensor nodes. The sensor nodes, which are intended to be physically small and inexpensive, are equipped with one or more sensors, a short-range radio transceiver, a small micro-controller, and a power supply in the form of a battery.

Sensor network deployments are envisioned to be done in large scales, where each network consists of hundreds or even thousands of sensor nodes. In such a deployment, human configuration of each sensor node is usually not feasible and therefore self-configuration of the sensor nodes is important. Energy efficiency is also critical, especially in situations where it is not possible to replace sensor node batteries. Battery replacement maintenance is also important to minimize for deployments where battery replacement is possible.

Most sensor network applications aim at monitoring or detection of phenomena. Examples include office building environment control, wild-life habitat monitoring [17], and forest fire detection [24]. For such applications, the sensor networks cannot operate in complete isolation; there must be a way for a monitoring entity to gain access to the data produced by the sensor network. By connecting the sensor network to an existing network infrastructure such as the global Internet, a local-area network, or a private

intranet, remote access to the sensor network can be achieved. Given that the TCP/IP protocol suite has become the de-facto networking standard, not only for the global Internet but also for local-area networks, it is of particular interest to look at methods for interconnecting sensor networks and TCP/IP networks. In this paper, we discuss a number of ways to connect sensor networks to TCP/IP networks.

Sensor networks often are intended to run specialized communication protocols, thereby making it impossible to directly connect the sensor network with a TCP/IP network. The most commonly suggested way to get the sensor network to communicate with a TCP/IP network is to deploy a proxy between the sensor network and the TCP/IP network. The proxy is able to communicate both with the sensors in the sensor network and hosts on the TCP/IP network, and is thereby able to either relay the information gathered by the sensors, or to act as a front-end for the sensor network.

Delay Tolerant Networking (DTN) [9] is a recently proposed communication model for environments where the communication is characterized by long or unpredictable delays and potentially high bit-error rates. Examples include mobile networks for inaccessible environments, satellite communication, and certain forms of sensor networks. DTN creates an overlay network on top of the Internet and uses late address binding in order to achieve independence of the underlying bearer protocols and addressing schemes. TCP/IP and sensor network interconnection could be done by using a DTN overlay on top of the two networks.

Finally, by directly running the TCP/IP protocol suite in the sensor network, it would be possible to connect the sensor network and the TCP/IP network without requiring proxies or gateways. In a TCP/IP sensor network, sensor data could be sent using the best-effort transport protocol UDP, and the reliable byte-stream transport protocol TCP would be used for administrative tasks such as sensor configuration and binary code downloads.

Due to the power and memory restrictions of the small 8-bit micro-controllers in the sensor nodes, it is often assumed that TCP/IP is not possible to run in sensor networks. In previous work [8], we have shown that this is not true; even small micro-sensor nodes are able to run a full instance of the TCP/IP protocol stack. We have also successfully implemented our small uIP TCP/IP stack [7] on the small sensor nodes developed at FU Berlin [1]. There are, however, a number of problems that needs to be solved before TCP/IP can be a viable alternative for sensor network communication.

The rest of the paper is structured as follows. We discuss proxy architectures in Section 2, followed by a discussion of the DTN architecture in Section 3. TCP/IP for sensor networks is presented in Section 4, and a comparison of the three methods is given in Section 5. Finally, the paper is concluded in Section 6.

2 Proxy Architectures

Deploying a special proxy server between the sensor network and the TCP/IP network is a very simple and straightforward way to connect the two networks. In its simplest form, the proxy resides as a custom-made program running on a gateway computer which has access to both the sensor network and the TCP/IP network. Since all interaction

between clients in the TCP/IP network and the sensor nodes is done through the proxy, the communication protocol used in the sensor network may be chosen freely.

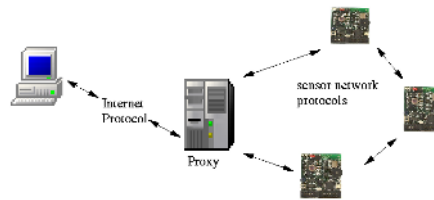


Fig. 1. Proxy architecture

The proxy can operate in either of two ways: as a relay, or as a front-end. In the first case, the proxy will simply relay data coming from the sensor network to clients on the TCP/IP network. The clients must register a particular *data interest* with the proxy, and the proxy will then relay data from the sensor network to the registered clients.

In the second case, where the proxy acts as a front-end for the sensor network, the proxy pro-actively collects data from the sensors and stores the information in a database. The clients can query the proxy for specific sensor data in a variety of ways, such as through SQL-queries or web-based interfaces.

One advantage of the proxy based approach to interconnect sensor and TCP/IP networks is that the proxy completely decouples the two networks. This naturally allows for specialized communication protocols to be implemented in the sensor network. A front-end proxy can also be used to implement security features such as user and data authentication.

Among the drawbacks of the proxy approach are that it creates a single point of failure. If the proxy fails, all communication to and from the sensor network is effectively made impossible. One possible solution would be to deploy redundancy in the form of a set of back-up proxies. Unfortunately, such a solution reduces the simplicity of the proxy approach. Other drawbacks are that a proxy implementation usually is specialized for a specific task or a particular set of protocols. Such a proxy implementation requires special proxies for each application. Also, no general mechanism for inter-routing between proxies exist.

Proxies have previously been used for connecting devices to TCP/IP networks in order to overcome limitations posed by the devices themselves, or limitations caused by the communication environment in which the devices are located. The Wireless Application Protocol (WAP) stack [15] is intended to be simpler than the TCP/IP protocol stack in order to run on smaller devices, and to be better suited to wireless environments. WAP proxies are used to connect WAP devices with the Internet. Similarly, the Remote Socket Architecture [23] exports the BSD socket interface to a proxy in order to outperform ordinary TCP/IP for wireless links.

3 Delay Tolerant Networks

The Delay Tolerant Network architecture [9] is intended for so-called *challenged environments*. Properties of such environments include long and variable delays, frequent network partitioning, potentially high bit-error rates and asymmetrical data rates. DTN is based on the observation that the TCP/IP protocol suite is built around a number of implicit assumptions that do not hold true in challenged communication environments. In particular, the underlying assumptions of TCP/IP are:

- An end-to-end path must exist between source and destination during the whole data exchange.
- The maximum round trip-time for packets must be relatively small and stable.
- The end-to-end packet loss is relatively small.

The DTN architectural design contains several principles to provide service in these environments:

- DTN uses an overlay architecture based on store-and-forward message switching. The messages, called *bundles*, that are transmitted contain both user data and relevant meta-data. A message-switched architecture provides the advantage of a priori knowledge of the size and performance requirements of the data transfer. The bundle layer works as an application layer on top the TCP/IP protocol stack.
- The base transfer between nodes relies on store-and-forward techniques, i.e., a packet is kept until it can be sent to the next hop. This requires that every node has storage available in the network. Furthermore, this allows to advance the point of retransmission towards the destination.

A DTN consists a set of *regions* which share a common layer called the *bundle layer* that resides above the transport layer. The bundle layer stores messages in persistent storage if there is no link available, fragments messages if necessary, and optionally implements end-to-end reliability. The layers below the bundle layer are not specified by the architecture, but are chosen dynamically based on the specific communication characteristics and the available protocols in each region. One or more DTN gateways exist in each DTN region. The DTN gateway forwards bundles between regions, and takes care of delivering messages from other regions to hosts within the local region.

The DTN architecture has been designed with the sensor network paradigm in mind. In sensor networks, the network may be partitioned frequently when nodes go into sleep mode or because of node failure. This will disrupt any end-to-end paths through the network. Also, packet loss rates in sensor networks can be very high [28] and routes may be asymmetric.

When connecting sensor networks to a TCP/IP network using the DTN architecture, we have at least two regions as depicted in Figure 2: one TCP/IP region where the TCP/IP protocol suite is used and one sensor network region where specialized sensor network protocols are implemented. A DTN gateway node is put in between the two networks, similar to where a proxy would have been placed.

The DTN gateway acts much as a relay proxy as discussed in the previous section, and the relay proxy approach can be viewed as a specific instance of the DTN architecture. The DTN architecture is much more general than a simple proxy based approach,

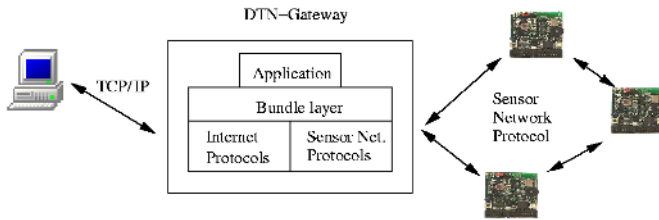


Fig. 2. Connecting using the DTN architecture

however, as the DTN architecture even allows mapping the sensor network into more than one DTN region, with DTN gateways located within the sensor network. For sensor networks where network partitioning is frequent, or where end-to-end communication is impossible, such a network design would be appropriate. A fully DTN enabled sensor network would easily be extended to a TCP/IP network, simply by connecting one or more of the DTN gateways to the TCP/IP network.

4 TCP/IP for Sensor Networks

Directly employing the TCP/IP protocol suite as the communication protocol in the sensor network would enable seamless integration of the sensor network and any TCP/IP network. No special intermediary nodes or gateways would be needed for connecting a sensor network with a TCP/IP network. Rather, the connection would simply be done by connecting one or more sensor nodes to the TCP/IP network. TCP/IP in the sensor network would also provide the possibility to route data to and from the sensor network over standard technologies such as General Packet Radio Service (GPRS) [4]. This leads to an architecture as shown in Figure 3.

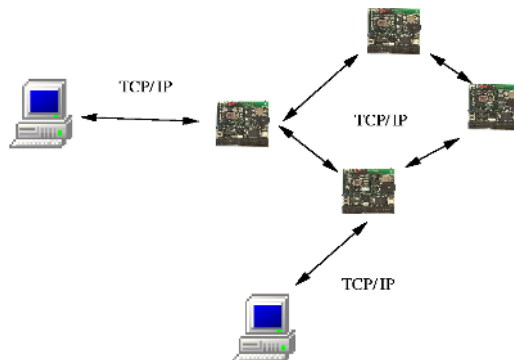


Fig. 3. Connecting using TCP/IP in the sensor network

Until recently, many believed that tiny sensor nodes would lack the necessary memory and computational resources to be able to run a full instance of the TCP/IP protocol stack. Therefore, the idea of using TCP/IP for sensor networks has not been given much research attention. We have showed that a full TCP/IP stack indeed can be run even on very limited devices [8], and have implemented our small uIP TCP/IP implementation [7] on the sensor nodes developed at FU Berlin [1]. These nodes are equipped with an 8-bit Texas Instruments MSP430 low-power micro-controller with a built-in memory of 2048 bytes. Our TCP/IP implementation requires only a few hundreds bytes of memory to operate, which leaves plenty of memory for the actual sensor node applications.

The fact that we are able to run the TCP/IP stack even on tiny sensor nodes suggest that TCP/IP for sensor networks may be within reach. Sensor networks running the TCP/IP protocol suite would be very easy to connect to existing TCP/IP networks, and would also be able to benefit from the wealth of readily available applications such as file transfers using FTP or HTTP and possibly time synchronization with NTP. There are, however, a number of problems with using TCP/IP for wireless sensor networks that need to be addressed before TCP/IP is a viable alternative for sensor networks:

- The addressing and routing schemes of IP are host-centric.
- The header overhead in TCP/IP is very large for small packets.
- TCP does not perform well over links with high bit-error rates, such as wireless links.
- The end-to-end retransmissions used by TCP consumes energy at every hop of the retransmission path.

IP is designed so that every network interface connected to a network has its own IP address. The prefix of the address is the same for all network interfaces in the same physical network and routing is done based on the network prefixes. This does not fit well with the sensor network paradigm, where the main interest is the data generated by the sensors and the individual sensor is of minor importance. Most of the proposed communication protocols for sensor networks use data centric routing and addressing [10, 12] and even though similar mechanisms have been developed as overlay networks on top of IP [21], these usually require too much state to be kept in the participating nodes to be feasible to run on limited sensor nodes.

The size of TCP/IP packet headers is between 28 and 40 bytes, and when sending a few bytes of sensor data in a datagram the headers constitute nearly 90% of each packet. Energy efficiency is of prime importance for sensor networks, and since radio transmission often is the most energy consuming activity in a sensor node [20], a header overhead of 90% is not acceptable. Hence, most protocols developed for sensor networks strive to keep the header overhead as low as possible. For example, the TinyOS [11] message header overhead is only 5%. The header overhead in TCP/IP can be reduced using various forms of header compression [13,6,16,5]. These mechanisms are commonly designed to work only over a single-hop link, but work is currently being done in trying to adopt these mechanisms to the multi-hop case [19].

Furthermore, since TCP was designed for wired networks where bit-errors are uncommon and where packet drops nearly always are due to congestion, TCP always interprets packet drops as a sign of congestion and reduces its sending rate in response

to a dropped packet. This leads to bad performance over wireless links where packets frequently are dropped because of bit-errors. TCP misinterprets the packet loss as congestion and lowers the sending rate, even though the network is not congested.

Also, TCP uses end-to-end retransmissions, which in a multi-hop sensor network requires a transmission by every sensor node on the path from the sender to the receiver. Such a retransmission consumes more energy than a retransmission scheme where the point of retransmission is moved closer to the receiver. Protocols using other mechanisms to implement reliability, such as reliable protocols especially developed for sensor networks [22,27,26], are typically designed to be energy conserving.

Methods for improving TCP performance in wireless networks have been proposed [2,3,14], but these are often targeted towards the case where the wireless link is the last-hop, and not for wireless networks with multiple wireless hops. In addition, traditional methods assume that the routing nodes have significantly larger amounts of resources than what limited sensor nodes have.

5 Comparison of the Methods

The three methods for connecting sensor networks to TCP/IP networks presented here are in some respects orthogonal—it is possible to make combinations such as a partially TCP/IP-based sensor network with a DTN overlay connected to the global Internet using an front-end proxy. It is therefore not possible to make a direct comparison of the methods. Instead, we will state the merits and drawbacks of each of the methods and comment on situations in which each method is suited.

A pure proxy method works well when the sensor network is deployed relatively close to a place where a proxy server can be safely placed. Since the proxy server by design must have more processing power and more memory than the sensors, it is likely to require an electrical power supply rather than a battery. Also, the proxy may need to be equipped with a stable storage media such as a hard disk, which may make the proxy physically larger than the sensor nodes. One example of a situation where these criteria are met is an office building environment. Here, a proxy server can be placed close to the sensor network, perhaps even in the same room as the sensors, and have immediate access to electrical power. Another example would be a nautical sensor network where the proxy could be equipped with a large battery pack and placed in the water with a buoy such that the significance of the physical size of the proxy node would be reduced.

Front-end proxies can also be used for a number of other things, besides for achieving interconnectivity, such as sensor network status monitoring, and generation of sensor failure reports to human operators.

The DTN architecture can be viewed as a generalization of the proxy architecture and indeed a DTN gateway shares many properties with a proxy server. A DTN gateway in the sensor network region will be placed at the same place as a proxy server would have been placed, and also requires more memory and stable storage media than the sensor nodes. There are, however, a number of things that are gained by using the DTN architecture rather than a simple proxy architecture. First, DTN inherently allows for multiple DTN gateways in a DTN region, which removes the single-point-of-failure problem of the simple proxy architecture. Second, while a proxy architecture usually is specialized

for the particular sensor network application, DTN provides general mechanisms and an interface that can be used for a large number of occasions. Also, if the sensor network is deployed in a place with a problematic communication environment, the DTN architecture provides a set of features which can be used to overcome the communication problems. Examples of such situations would be deep-sea exploration or places where seismic activity can disrupt communication.

From an interconnectivity perspective, running native TCP/IP in the sensor networks is the most convenient way to connect the sensor network with a TCP/IP network. One or more sensor nodes would simply be attached to the TCP/IP network, and the two networks could exchange information through any of those nodes. The attachment can be done either using a direct physical link, such as an Ethernet cable, or over a wireless technology like GPRS.

While a TCP/IP enabled sensor network may provide the easiest way to interconnect the networks, it is usually not a complete solution, but must be integrated into a larger architecture. The proxy and DTN architectures discussed here are examples of such an architecture. We can e.g. imagine an office building TCP/IP sensor network that is connected to a front-end proxy located in the cellar of the building. The connection between the proxy and the sensor network would be made using the regular TCP/IP local-area network in the building. Another example would be a TCP/IP sensor network for monitoring the in-door environment in a train. A DTN gateway would be placed in the same train, and the sensor network and the gateway would communicate using TCP/IP over the train's local area network. The DTN gateway would be able to transmit the gathered information over the global Internet at places where the train has Internet access.

Finally, from a security perspective, the front-end proxy architecture provides a good place to implement user and data authentication, since all access to the sensor network goes through the proxy. The DTN architecture is inherently designed for security and uses asymmetric cryptography to authenticate both individual messages and routers. TCP/IP as such does not provide any security, so security must be implemented externally either by using a front-end proxy, DTN, or any of the existing security mechanisms for TCP/IP networks such as Kerberos. It should also be noted that security methods developed especially with wireless sensor networks in mind [18,25] can be implemented as application layer security in TCP/IP sensor networks.

6 Conclusions

We have presented three methods for connecting wireless sensornets with TCP/IP networks: proxy architectures, Delay Tolerant Networking (DTN) overlays, and TCP/IP for sensor networks. The three methods are orthogonal in that it is possible to form combinations, such as a DTN overlay on top of a TCP/IP sensor network behind a front-end proxy.

The proxy architectures are simple and make it possible to use specialized communication protocols in the sensor network, but are application specific and creates a single point of failure. The DTN architecture also allows for specialized protocols, but

provides a much more general communication architecture. DTN is also useful if the sensor network itself is deployed in a challenged communication environment.

Finally, by using the TCP/IP protocol suite for the sensor network, connecting the sensor network with another TCP/IP network is simply done by attaching one or more sensor nodes to both networks. However, attaching the sensor nodes to the TCP/IP network may not always be ideal, and a combination of either a proxy architecture and TCP/IP, or DTN and TCP/IP, may be beneficial.

TCP/IP for sensor networks currently has a number of problems, and therefore further research in the area is needed before TCP/IP can be a viable alternative for sensor networking.

References

1. CST Group at FU Berlin. Scatterweb Embedded Sensor Board. Web page. 2003-10-21.
URL: <http://www.scatterweb.com/>
2. H. Balakrishnan, S. Seshan, E. Amir, and R. Katz. Improving TCP/IP performance over wireless networks. In *Proceedings of the first ACM Conference on Mobile Communications and Networking*, Berkeley, California, November 1995.
3. K. Brown and S. Singh. M-TCP: TCP for mobile cellular networks. *ACM Computer Communications Review*, 27(5):19–43, October 1997.
4. J. Cai and D. Goodman. General packet radio service in GSM. *IEEE Communications Magazine*, 35:122–131, October 1997.
5. S. Casner and V. Jacobson. Compressing IP/UDP/RTP headers for low-speed serial links. RFC 2508, Internet Engineering Task Force, February 1999.
6. M. Degermark, M. Engan, B. Nordgren, and S. Pink. Low-loss TCP/IP header compression for wireless networks. *ACM/Baltzer Journal on Wireless Networks*, 3(5), 1997.
7. A. Dunkels. uIP - a TCP/IP stack for 8- and 16-bit microcontrollers. Web page. 2003-10-21.
URL: <http://dunkels.com/adam/uiip/>
8. A. Dunkels. Full TCP/IP for 8-bit architectures. In *Proceedings of The First International Conference on Mobile Systems, Applications, and Services (MOBISYS '03)*, May 2003.
9. K. Fall. A delay-tolerant network architecture for challenged internets. In *Proceedings of the SIGCOMM'2003 conference*, 2003.
10. J. S. Heidemann, F. Silva, C. Intanagonwiwat, R. Govindan, D. Estrin, and D. Ganesan. Building efficient wireless sensor networks with low-level naming. In *Symposium on Operating Systems Principles*, pages 146–159, 2001.
11. J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for networked sensors. In *Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems*, November 2000.
12. C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *Mobile Computing and Networking*, pages 56–67, 2000.
13. V. Jacobson. Compressing TCP/IP headers for low-speed serial links. RFC 1144, Internet Engineering Task Force, February 1990.
14. J. Liu and S. Singh. ATCP: TCP for mobile ad hoc networks. *IEEE Journal on Selected Areas in Communications*, 19(7):1300–1315, 2001.
15. Wireless Application Protocol Forum Ltd. *Official Wireless Application Protocol: The Complete Standard*. Wiley Computer Publishing, 2000.
16. S. Pink M. Degermark, B. Nordgren. IP header compression. RFC 2507, Internet Engineering Task Force, February 1999.

17. A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson. Wireless sensor networks for habitat monitoring. In *First ACM Workshop on Wireless Sensor Networks and Applications (WSNA 2002)*, Atlanta, GA, USA, September 2002.
18. A. Perrig, R. Szewczyk, V. Wen, D. E. Culler, and J. D. Tygar. SPINS: security protocols for sensor networks. In *Mobile Computing and Networking*, pages 189–199, 2001.
19. S. Mishra R. Sridharan, R. Sridhar. A robust header compression technique for wireless ad hoc networks. In *MobiHoc 2003*, Annapolis, MD, USA, June 2003.
20. V. Raghunathan, C. Schurgers, S. Park, and M. Srivastava. Energy aware wireless microsensor networks. *IEEE Signal Processing Magazine*, 19(2):40–50, March 2002.
21. S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content addressable network. In *Proceedings of ACM SIGCOMM 2001*, 2001.
22. Y. Sankarasubramaniam, O. Akan, and I. Akyildiz. ESRT : Event-to-Sink Reliable Transport in Wireless Sensor Networks. In *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking and computing (MobiHOC 2003)*, 2003.
23. M. Schläger, B. Rathke, A. Wolisz, and S. Bodenstein. Advocating a remote socket architecture for internet access using wireless lans. *Mobile Networks and Applications*, 6(1):23–42, 2001. ISSN: 1383-469X
24. S. N. Simic and S. Sastry. Distributed environmental monitoring using random sensor networks. In *Proceedings of the 2nd International Workshop on Information Processing in Sensor Networks*, pages 582–592, Palo Alto, California, 2003.
25. F. Stajano. *Security for Ubiquitous Computing*. John Wiley and Sons, February 2002. ISBN: 0-470-84493-0
26. F. Stann and J. Heidemann. RMST: Reliable Data Transport in Sensor Networks. In *Proceedings of the First International Workshop on Sensor Net Protocols and Applications*, pages 102–112, Anchorage, Alaska, USA, April 2003. IEEE.
27. C.Y. Wan, A. T. Campbell, and L. Krishnamurthy. PSFQ: A Reliable Transport Protocol For Wireless Sensor Networks. In *First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA 2002)*, Atlanta, September 2002.
28. J. Zhao and R. Govindan. Understanding packet delivery performance in dense wireless sensor networks. In *The First ACM Conference on Embedded Networked Sensor Systems (SenSys 2003)*, Los Angeles, California, November 2003.

Experimental Analysis of Heterogeneous Wireless Networks

Giulio Iannello¹, Antonio Pescapè², Giorgio Ventre², and Luca Vollero²

¹ Campus Biomedico di Roma (Italy)
iannello@unina.it

² Università di Napoli "Federico II" (Italy)
{pescapè, giorgio, vollero}@unina.it

Abstract. Packet loss and delay in Internet degrade the quality of requested services like VoIP (Voice over IP) or Video Streaming. In novel network scenarios where wired and wireless connections are melted together, a real measure of these parameters is fundamental in a planning process of new services over novel network infrastructures. Nowadays networks are heterogeneous in terms of access network technologies (wired LAN Ethernet 10/100/1000, Wireless LAN - 802.11a, 802.11b, 802.11g -, GPRS, UMTS, GSM, Bluetooth, ...), end-users' devices (workstation, PC desktop, Laptop/Notebook, PDA, Advanced Mobile Phone, ...) and finally operating systems (Unix, Linux, Win 98/NT/2000/XP, Win CE, Linux Familiar, OS Embedded, ...). In this work we provide a heterogeneous network performance characterization with respect to delay and throughput in UDP and TCP environments. In order to determine our results we use an innovative traffic generator named D-ITG (Distributed Internet Traffic Generator). Results presented in this paper can be used as performance references for development of wireless communication applications over multiservice and heterogeneous networks.

Keywords: Heterogeneous networks, wireless networks performance analysis.

1 Introduction

In the last years network capacity has increased at a dramatic rate. At the same time the proliferation of the web has resulted in an exponential increase in the number of "surfing users" supported by the Internet. These users are becoming increasingly sophisticated and demand high-bandwidth, low-delay network services at affordable prices. These services' request is made on new "heterogeneous integrated and mobile" networks. In fact, as technology continues its dramatic progress, making possible new and improved applications, we experience the creation of new paradigms and changes in the way technology impacts every day's life.

Always-on connectivity, location-awareness, and environment-aware products are among these new paradigms. Smart devices, portable devices, wireless communications appear to be the underlying principles of a new revolution in technology. Pervasive computing deals with a wide range of information access methods enabled by mobility, wireless, small embedded systems, and broadband technologies [1]. Integration of fixed and portable wireless access to IP networks presents a cost effective and efficient way

to provide seamless end-to-end connectivity and ubiquitous access in a market where demands of mobile Internet have grown rapidly and predicted to generate billions of dollars in revenue.

This work provides a performance characterization of a real heterogeneous scenario where wireless and wired connections and where a wide range of end user device are present: in our real scenario we use PDA (Personal Digital Assistant), notebook/laptop, PC desktop and finally workstation. As far as this whole of end systems there is a wide range of operating systems present in our scenario. Measures were carried out on a testbed which reproduces (on a small scale) a real prototype of a heterogeneous mobile and integrated network. The study of the network behavior has been realized using D-ITG (*Distributed Internet Traffic Generator*) which provides a set of powerful tools for traffic patterns generation and results analysis. We present our experimental results and at the same time we analyze and validate theoretical assumptions on wireless performance behavior carried out in [2].

The paper is organized in 6 sections. After this introduction, in the next section the motivations and the reference framework on which our work is based are presented. Functionalities and main concepts regarding the D-ITG platform are shown in section 3. The experimental setup where our work has been carried out is presented in section 4, discussing the main issues related to our heterogeneous scenario and describing the measuring procedure. Section 5 reports the obtained experimental results. Finally, section 6 provides some concluding remarks and issues for research.

2 Motivation and Related Work

One of the most innovative concept and, at the same time, the most difficult challenge for all network engineers is actually that of “integration”: a unique and pervasive network scenario for the support of all the traffic (data, voice and video). A unique infrastructure but, above all, a unique protocol, the protocol IP, glue of all applications on different platforms: situations in which wireline world and wireless world are melted together are nowadays realities and the actual trend is the definition of a global communication paradigm, independent from the particular network technologies.

Among the many innovations introduced in the IP networks, an interesting challenge is to bring services like telephony and video transmission on the same infrastructure used for data traffic. This process relies on using QoS (*Quality of Service*) approach and at same time on the precise characterization of used heterogeneous network scenario. For these reasons, performance and experimental analysis of wireless networks is currently an important research issue. There are several simulation and analytical studies on wireless channel performance [15] [16], whereas in this work, we test a real heterogeneous mobile environment made by heterogeneous (wired and wireless) network, heterogeneous users’ device (Laptop, PDA, Advanced Mobile Phone, Workstation,...) and finally heterogeneous operating system. Our scenario is heterogeneous in terms of:

- access network technologies (*WLAN 802.11, wired Ethernet 10/100 Mbps*)
- end-users’ devices (*PDA, Laptop, PC desktop*)
- end-users’ operating systems (*Linux Embedded, Linux, Windows XP/ 2k/CE*)

Over this heterogeneous scenario we carried out a complete performance study of a real heterogeneous and integrated mobile network. In a situation where a roaming user

sends traffic both to another roaming user and to a fixed position, experimental results on throughput and delay (using both UDP and TCP connections) are presented. We assess our results showing the different performance (between roaming end-nodes) at different distances.

Before presenting our results and in order to provide a general framework a brief state of the art related to other similar works is presented. A performance characterization of ad hoc wireless networks is presented in [3]. The paper examines impact of varying packet size, beaconing interval, and route hop count on communication throughput, end-to-end delay, and packet loss. In [4] a new performance model for the IEEE 802.11 WLAN in ad hoc mode is presented. Three adjustable parameters are presented: packet fragmentation factor, buffer size, and maximum allowable number of retransmissions. In the work there is the measure the system performance by using three parameters: throughput, delay, and probability of fail to deliver.

In [5], three techniques for composite performance and availability analysis are discussed in detail through a queuing system in a wireless communication network. In [6] there is a study on network performance of commercial IEEE 802.11 compliant WLANs measured at the MAC sublayer in order to characterize their behavior in terms of throughput and response time under different network load conditions.

A performance study on wireless LAN in a vehicular mobility scenario is presented in [7]. In [8] the performance of a real campus area network are measured. In order to carry out the results the authors used three performance monitoring software: CWINS Wireless Benchmarking tool, Harris LAN Evolution Software and WaveLan Diagnostic Software. Performance measuring has been carried out moving on several parameters: received power, walls and floors separating two radio interfaces and finally interfering traffic. In [9] the authors present a comprehensive study on TCP and UDP behavior over WLAN taking into account radio hardware, device drivers and network protocols. [10] presents a performance measurements carried out on a real MAN in order to apprehend the real throughput.

3 Distributed Internet Traffic Generator (D-ITG)

The successful evolution of network research is tightly coupled to the ability to design simple and accurate models with the propriety (and possibility) of traffic patterns reproducibility. Traffic theory suggest us the application of mathematical modeling to explain traffic performance relationship with network capacity, traffic demand and experimented performance.

One of the applications of traffic models is the generation of synthetic, yet realistic traffic to be injected into a network, in order to simulate the behavior of a multitude of real traffic sources. In the case of studies related to the Internet, simulations should reflect not only the wide scale of real scenarios, by also the rich variety of traffic sources, in terms both of protocol typologies and of data generation patterns.

The purpose of the *Distributed Internet Traffic Generator* [11] [12] [13] is to build up a suite that can be easily used to generate repeatable sets of experiments by using a reliable and realistic mixture of available traffic typologies. By using configurable scenario procedures on individual machines, and by coordinating the actions of these network devices, with D-ITG is possible to generate many traffic test-cases that could

be originated by a typical network scenario made of large number of users and network devices, as well as by different network topologies.

We believe that D-ITG shows interesting properties when compared to other traffic generators. A centralized version and two kinds of distributed generators have been implemented. In the first distributed version there is a log server that is used by senders and receivers for data logging. Both communications between senders and log server and receiver and log server are carried out using both TCP/IP transport protocols: UDP and TCP. In the second distributed version, processes of both senders and receivers have been implemented using MPI library [14]. By separating generation and log processes, it has been eliminated the interference problem between them, which results in better overall performance.

By eliminating interference problems the distributed version is able to replicate theoretical traffic figure imposed at sender side with greater accuracy. Furthermore in a heterogeneous mobile scenario made by communications between PDA or PocketPC, using this distributed version it is possible to generate high traffic rate on the mobile device and at the same time to log sent and received traffic on a server present in the wired network: this modus operandi provide an alternative way to data logging on device where the storage capacity is very small.

Due to the nodes' limited resource (RAM, storage capacity, video dimension, ...) in wireless ad hoc networks, scalability is crucial for network operations. In particular a distributed approach to network communication using collaborative mechanism permits reaching comparable performance respect to wired scenario. Indeed using a log server for sender and receiver logging phase we can assure greater performance when we use PocketPc too.

In order to carry out a complete characterization of heterogeneous integrated and mobile networks D-ITG has been ported on several different operating systems: Linux, Windows, and embedded operating systems. With respect to this last platform in our testbed we used PDAs where is running the Linux FAMILIAR - kernel 2.4.18 version, and original source code, with little modifications, has been ported on this destination platform using a cross-compiler version of gcc. Currently we are working on a porting on WinCE platform too.

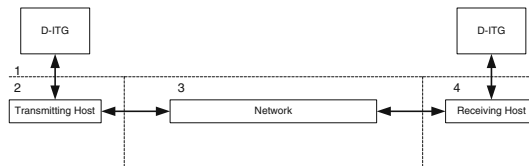


Fig. 1. Testbed Schema

4 Experimental Setup Description

The goal of our analysis is the performance characterization of heterogeneous networks in which wireless links are present. In order to pursue this objective a set of experimental setups with similar characteristics has been chosen. All tests can be collapsed in a same general scenario, depicted in figure 1, where two communication entities, a D-ITG transmitter and a D-ITG receiver, are directly connected through an IP network channel. Indeed, as represented in figure 1, the tests differ for the type of used network (3), its configuration (3) and the type of host (2-4) that executes the D-ITG platform. Others parametric elements, like generated traffic patterns, have not been changed, using only periodical sources, with fixed packet size (PS) and fixed inter-departure times (IDT) between packets. In table 1 the complete set of parametric elements used in our tests is summarized. In the case of ad-hoc configuration, we have experimented more situations, allowing to the two communicating hosts to move at various mutual distances.

The characterization has been carried out for both IP transport protocols (UDP and TCP) in three different traffic conditions:

- *low* traffic load ($\leq 1.2Mbps$)
- *medium* traffic load ($\leq 4.0Mbps$)
- *high* traffic load ($\leq 5.12Mbps$)

These three traffic conditions are related to three different real traffic loads and at the same time three different load conditions in theoretical wireless channel models. For every traffic condition, we have analyzed three type of hosts configuration: (i) classic configuration, with only laptop and workstation devices, (ii) pocket receiving configuration, where the receiving host is always a PocketPC, and (iii) pocket transmitting configuration, where the transmitting host is always a PocketPC.

Table 1. Parametric Elements

Testbed Element	Description	Set of Elements
1 - D-ITG	Protocol	{UDP, TCP}
	IDT	$\{\frac{1}{100}, \frac{1}{1000}, \frac{1}{10000}\}$ s
	Packet Size	{64, 128, 256, 512, 1024, 1500} bytes
2 - Tx-Host	Typology of host	{Workstation, Laptop, PocketPC}
3 - Network	Network typology and configuration	{Wired-Wired, Wired-Wireless, Wireless-Wireless, With or Without Access Point(AP), ...}
4 - Rx-Host	Typology of host	{Workstation, Laptop, PocketPC}

In order to characterize a system like that one depicted in figure 1, measured metrics are the (source/destination)-bandwidth (UDP and TCP protocols) and the delay deviation (mean and standard deviation) with respect to the time of the first received packet (UDP only). For each measured parameter, several trials have been performed in the same operating conditions.

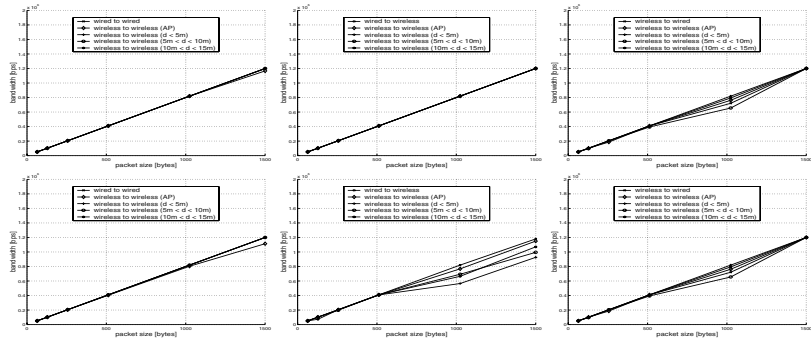


Fig. 2. UDP transmission(top)/receiving(bottom) bandwidth for $IDT = \frac{1}{100}$ s

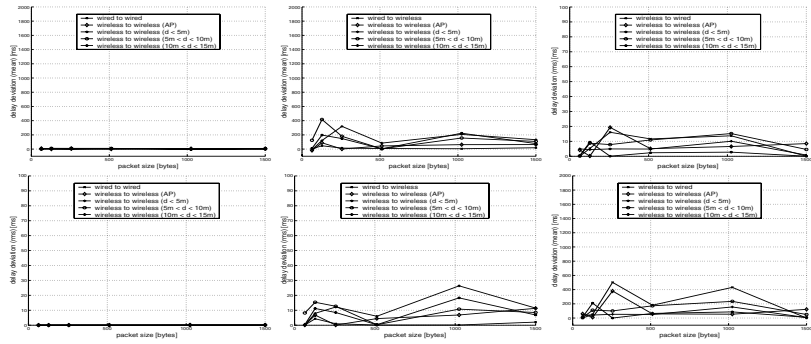


Fig. 3. Mean(top) and standard deviation(bottom) of the delay deviation for $IDT = \frac{1}{100}$ s

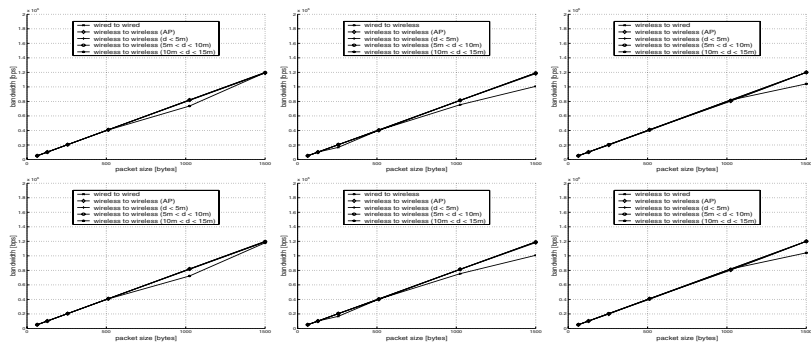


Fig. 4. TCP transmission(top)/receiving(bottom) bandwidth for $IDT = \frac{1}{100}$ s

Table 2. Picture Legend

Legend	Description
wired-to-wired	Connection between two workstation through Ethernet 10/100 Mbps network
wired-to-wireless	Connection between the workstation and the laptop/pocketPC through AP
wireless-to-wireless (AP)	Connection between laptop and pocketPC through AP
wireless-to-wireless ($d \leq x$)	Connection between laptop and pocketPC in ad-hoc mode in a range of x meters

Table 3. Technical details on experimental setup

Host/Device	Description
Laptop1	IBM T23, Mobile Intel PIII 1133 Mhz, Main Memory 128 MB, Cache 256 KB, O.S. Linux Red Hat 9.0 – kernel 2.4.20-18.9
Laptop2	Acer TravelMate 351 TE: PIII 700 Mhz, Main Memory 128 MB
Workstation1	PC sender, Intel PII 850 Mhz, Main Memory 128 MB, Cache 256 KB, dual boot Operating Systems: Linux Red Hat 7.1 – kernel 2.4.2-2, Windows XP Professional Service Pack 1
Workstation2	PC receiver, Intel C 400 Mhz, Main Memory 64 MB, Cache 128 KB, O.S. Linux Red Hat 7.0 - kernel 2.4.2-19
PocketPC	Compaq iPAQ H3850, Intel StrongARM 206 Mhz, Main Memory 64 MB, Flash ROM 32 MB, O.S. Linux FAMILIAR – kernel 2.4.18
Access Point	Orinoco Ap1000, 11Mbps (802.11b), Multi Channel support
Wireless LAN cards	WiFi ORINOCO 11Mbps GOLD

5 Performance Analysis and Experimentation

In this section we present measures obtained in the various cases. We step from showing and analyzing the results for *low* load traffic condition, then we present the results for *medium* and finally for *high* traffic load. In table 3 details on used devices are depicted.

In next figures we show comparative analysis on mobile environment using roaming user in three stage of space ($d \leq 5 m$, $5 m \leq d \leq 10 m$, $10 m \leq d \leq 15 m$). In the three different traffic conditions we use different packet size dimensions. Indeed in the first traffic profile we can use a packet size dimension up to 1500 bytes (according to *low* traffic load). In the second traffic profile we use a packet size dimension up to 512 bytes (according to *medium* traffic load). Finally in the third traffic profile we use only one packet size dimension equal to 64 bytes (according to *high* traffic load).

5.1 Low Traffic Load

Test results for *low* traffic load are depicted in figures 2, 3 and 4. For *low* traffic load we mean a traffic state in which we are far from the saturated wireless channel condition.

The sending/receiving bandwidth is reported in figures 2 and 4, using respectively UDP and TCP transport protocols. Instead, in figure 3 the behavior of the delay deviation with respect to the time of the first received packet is reported.

First row of figures 2 and 4 represents the behavior observed by the transmitting host, while the second one represents the behavior observed by receiving host. The first row of figure 3 shows mean delay deviation behavior, while the second one represents the delay standard deviation for all considered configuration. In all these figures, the left columns is related to a situation in which the communication entities are two workstations, or one workstation and one laptop; instead, the others two columns are related to a scenario in which the transmitter (right) or the receiving (center) host is always a PocketPC, while the transmitting/receiving one can be a workstation (wired element) or a laptop (wireless element). In table 2 the complete reference for the legend used in these and in the following graphs is reported.

In order to have a reference curve, it has been generated also the diagram related to direct wired connection, in the *workstation to workstation* classical configuration. From the bandwidth diagrams produced for the several configurations, two aspects are clearly depicted: (i) the communication is reliable and (ii) the degradation of the performance is due to the smaller computational power of the adopted devices (PDAs). It is interesting to notice that TCP suffers the losses mainly, having a different behavior with respect to UDP; TCP indeed interprets the losses like due to congestion phenomena and reacts consequently, reducing the maximum transmittable rate and emphasizing the phenomenon of bandwidth reduction. Of particular interest is the case of 1500 bytes packets, where the packet dimension exceeds MTU (Maximum Transfer Unit), the maximum allowable dimension of a MAC data unit. The fragmentation produces the duplication of the total number of transmitted packet and it exacerbates the throughput reduction of the wireless channel.

Analysis of the delay diagrams demonstrate that the strong sensitivity of the delay deviation is function of the used configuration and the used hosts: when a wireless link is used, the arrival time of the first packet is little meaningful respect to the total delay. For this reason a measure of mean and standard deviation is useful. Moreover, the delay diagrams also demonstrate the uncorrelation between the perceived bandwidth and the packet delay.

5.2 Medium Traffic Load

The test results for *medium* traffic load are depicted in figures 5, 6 and 7. For *medium* traffic load we mean a traffic state in which we are closed the saturated wireless channel condition. In order to quantify the proximity to the saturated channel condition, in the diagrams of the throughput it has been brought back also the diagram obtained from the Bianchi theoretical model [2]. In [2] a simple analytical model to compute the saturation throughput performance of the 802.11 is presented. The model assumes a finite number of terminals and ideal channel conditions and it is suited for any access scheme employed. The model shows that performance of the basic access method strongly depends on the system parameters, mainly packet size dimension and number of stations in the wireless network. Such model gives us a bound to the maximum traffic load that can cross the channel at the MAC layer of the ISO/OSI stack, therefore it supplies a useful bound

for the traffic at the upper layer. Using our experimental results, we can also provide a practical validation of the Bianchi theoretical model.

Diagrams organization is equal to the that one present in the subsection 5.1. In this load condition it turns out with more evidence the dependency from the host typology and the used transport protocol. TCP still demonstrates of being more sensitive to the losses respect to UDP. However, regarding the previous case we can observe the greater sensitivity respect to packet dimension of the wireless configurations, especially of those with PocketPC.

The delay diagrams confirm that the strong sensitivity of the delay deviation in function of the used configuration and the used hosts, and prove, when a wireless link is used, that the arrival time of the first packet is little meaningful of the total delay. However, regarding the previous case we can observe a greater tie between the observed throughput reduction and delay variations.

5.3 High Traffic Load

Test results for *high* traffic load are depicted in figures 8, 9 and 10. For *high* traffic load we mean a traffic state in which we are in the saturated wireless channel condition, and every station has always a packet ready for the transmission.

With respect to previous cases we have analyzed only a transmission condition where the packet size is equal to 64 bytes. Indeed, for whichever packet dimension the channel turns out saturated: longer packets carry to a greater channel busy time for delivered or collided packet, and it only leads to a greater number of losses from the sender side for network interface saturation. The organization of the diagrams is the same one of the previous cases, the only difference is in having brought back the transmission and reception plots in the same area using histogram diagrams (in this case we have changed the figures model because we have only one packet dimension).

It is interesting to notice the behavior of UDP and TCP in the several analyzed configurations: TCP reacts to the saturation condition limiting the demanded transmission bandwidth, while UDP endures a highest packet loss. This behavior is caused from the presence of a flow-control mechanism in the first protocol, and from the ability to the congestion control of TCP to optimize the use of a high loaded channel.

6 Conclusions and Issues for Research

In this work we presented a general framework for traffic analysis and performance characterization in heterogeneous mobile networks. This work steps from the assumption that a current realistic scenario must consider the fusion of wired and wireless connection and several kinds of user devices. A number of tests conducted on our real testbed yielded important characteristics such as throughput and delay under various network loads. Our results demonstrate, in the low traffic load situation, the uncorrelation between the perceived bandwidth and the packets delay. In the other two traffic load situations we observed a greater tie between the observed throughput reduction and delay variations.

The paper presents a complete experimental analysis in UDP and TCP scenarios with respect to throughput and delay. The fundamental contribution of our work was the clear definition of which system's elements are responsible of network performances

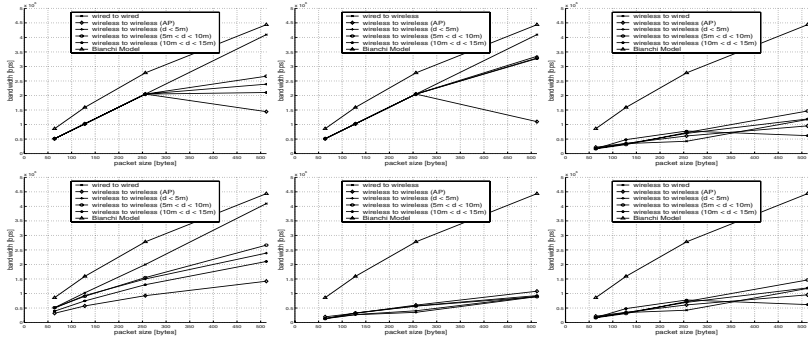


Fig. 5. UDP transmission(top)/receiving(bottom) bandwidth for $IDT = \frac{1}{1000}$ s

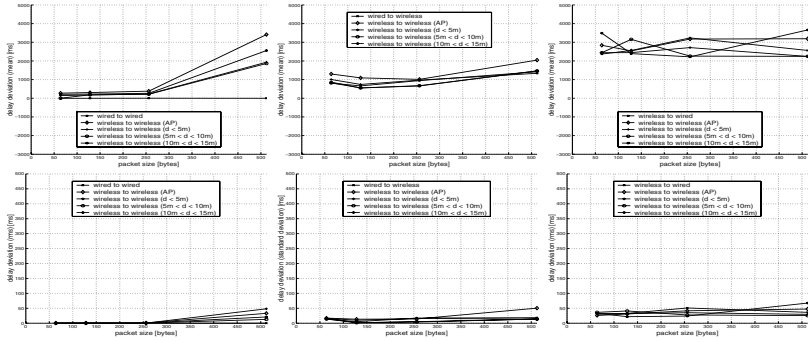


Fig. 6. Mean(top) and standard deviation(bottom) of the delay deviation for $IDT = \frac{1}{1000}$ s

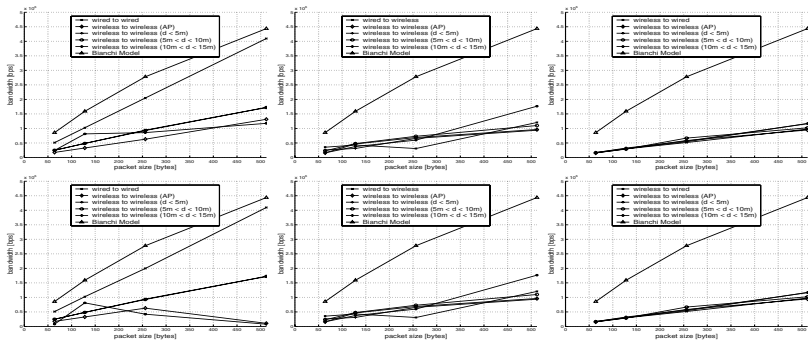


Fig. 7. TCP transmission(top)/receiving(bottom) bandwidth for $IDT = \frac{1}{1000}$ s

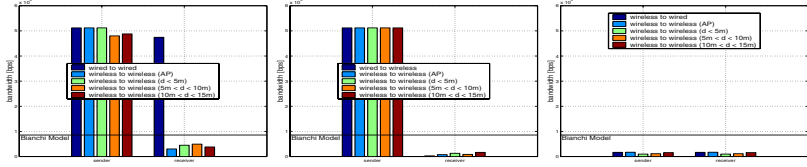


Fig. 8. UDP transmission/receiving bandwidth for $IDT = \frac{1}{10000}$ s

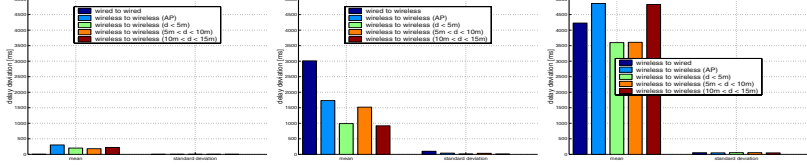


Fig. 9. Mean and standard deviation of the delay deviation for $IDT = \frac{1}{10000}$ s

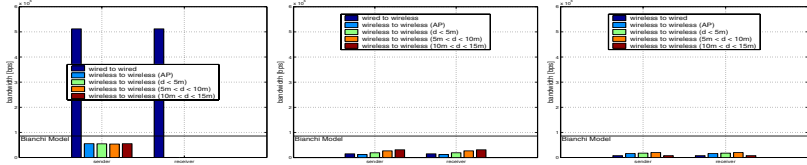


Fig. 10. TCP transmission/receiving bandwidth for $IDT = \frac{1}{10000}$ s

degradation and how to use different protocols impacts observed on the traffic behavior. Furthermore, using our results the analytical model presented by Bianchi [2] is validated. We have demonstrated that it is useful as a upper bound reference throughput in the case of real traffic scenarios.

Results showed in this work can be used as references for development of wireless communication applications. Indeed in a planning phase of innovative applications over heterogeneous networks is necessary a complete parametric network characterization. Currently, our testbed allows experiments on a small-scale. We will test the system behavior on a realistic network of a much wider-scale. Furthermore we are working on a similar analysis presented in this paper in a scenario where interference due to Bluetooth and IrDA communications are present. Finally, using D-ITG capabilities we will test a similar scenario using different traffic patterns made by different stochastic IDT and PS distributions according to several theoretical traffic models.

Acknowledgments. This work has been carried out partially under the financial support of the “Ministero dell’Istruzione, dell’Università e della Ricerca (MIUR)” in the framework of the FIRB Project “Middleware for advanced services over large-scale, wired-wireless distributed systems (WEB-MINDS)”.

References

1. J. P. Munson, M. Viveros, "Pervasive and Mobile Commerce Applications" *IPCCC 2002 – 21st IEEE International Performance, Computing, and Communications Conference*, April 2002 – Phoenix, Arizona
2. G. Bianchi, "Performance Analysis of the 802.11 Distributed Coordination Function", *Selected Area in Communications, IEEE Journal on* Volume: 18 Issue: 3, March 2000 Page(s): 535–547
3. C. -K. Toh, M. Delwar, D. Allen, "Evaluating the communication performance of an ad hoc wireless network", *Wireless Communications, IEEE Transactions on* , Volume: 1 Issue: 3 , July 2002 Page(s): 402–414
4. F. Eshghi, A. K. Elhakeem, "Performance analysis of ad hoc wireless LANs for real-time traffic", *Selected Areas in Communications, IEEE Journal on* , Volume: 21 Issue: 2 , Feb. 2003 Page(s): 204–215
5. Ma Yue, J. J. Han, K. S. Trivedi, "Composite performance and availability analysis of wireless communication networks", *Vehicular Technology, IEEE Transactions on* , Volume: 50 Issue: 5 , Sept. 2001 Page(s): 1216–1223
6. B. Bing, "Measured performance of the IEEE 802.11 wireless LAN", *Local Computer Networks, 1999. LCN '99. Conference on* , 18–20 Oct. 1999 Page(s): 34–42
7. J. P. Singh, N. Bambos, B. Srinivasan, D. Clawin., "Wireless LAN performance under varied stress conditions in vehicular traffic scenarios", *Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall*. 2002 IEEE 56th , Volume: 2 , 24–28 Sept. 2002 Page(s): 743–747 vol.2
8. A. Messier, J. Robinson, K. Pahlavan, "Performance monitoring of a wireless campus area network", *Local Computer Networks, 1997. Proceedings, 22nd Annual Conference on* , 2–5 Nov. 1997 Page(s): 232–238
9. G. Xylomenos, G. C. Polyzos, "TCP and UDP performance over a wireless LAN", *INFOCOM '99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE* , Volume: 2 , 21–25 March 1999 Page(s): 439–446 vol.2
10. J. C. Amaro, R. P. Lopes, "Performance analysis of a wireless MAN", *Network Computing and Applications, 2001. NCA 2001. IEEE International Symposium on*, 8–10 Oct. 2001 Page(s): 358–361
11. A. Pescapè, M. D'Arienzo, S. P. Romano, M. Esposito, S. Avallone, G. Ventre, "Mtools" – IEEE Network – Software Tools for Networking – September/October 2002, Vol. 16 No. 5 pag. 3. ISSN 0890–80445
12. A. Pescapè, S. Avallone, G. Ventre "Analysis and experimentation of Internet Traffic Generator", accepted to New2an'04
13. A. Pescapè, S. Avallone, G. Ventre "Distributed Internet Traffic Generator (D-ITG): analysis and experimentation over heterogeneous networks", poster at ICNP 2003 <http://icnp03.cc.gatech.edu/posters.html>, accessed on Nov. 2003
14. Message Passing Interface Forum, "MPI: A Message-Passing Interface Standard", *International Journal of Supercomputer Applications*, Vol.3, No.4, pp 165–414, August 1994.
15. M. Carvalho and J.J. Garcia-Luna-Aceves, "Delay Analysis of IEEE 802.11 in Single-Hop Networks" *Proc. IEEE ICNP 03: 11th IEEE International Conference on Network Protocols*, Atlanta, Georgia, November 4–7, 2003.
16. H. Wu, Y. Peng, K. Long, S. Cheng, J. Ma, "Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LAN: Analysis And Enhancement", *IEEE INFOCOM'2002*.

High-Level Behavioral SDL Model for the IEEE 802.15.3 MAC Protocol

Daniel Dietterle, Irina Babanskaja, Kai Dombrowski, and Rolf Kraemer

IHP, Im Technologiepark 25, D-15236 Frankfurt(Oder), Germany
{dietterle,babanskaja,dombro,kraemer}@ihp-
microelectronics.com

Abstract. In this paper, we present our behavioral SDL model for the IEEE 802.15.3 MAC protocol. The model was derived using an object-oriented approach based on the client/server paradigm and is divided into data path and control path subsystems. Within the model, there are different abstraction layers, each layer providing a well-defined set of services to the higher layers. This modular design approach facilitated teamwork and led to a model, which is understandable, easy to extend, adapt, and test. Thus, our SDL model can serve as a basis for the following steps in the design flow, which is also presented briefly.

Keywords: System design, wireless protocols, SDL, IEEE 802.15.3

1 Introduction

In recent years, wireless local area networks (WLANs) and wireless personal area networks (WPANs) have gained more and more importance as they provide connectivity for mobile devices in home or office environments. This development has been driven by a number of standardization efforts, with the IEEE 802.11 standards [1] and Bluetooth [2], now also an IEEE standard, being the most successful ones in the market.

A focus of our research activity is to design a low-power wireless communication system for mobile battery-powered devices and sensors. It shall provide transmission of asynchronous data, as well as audio and MPEG-1 encoded video streams. In other words, the communication system must provide a data throughput of 3-5 Mbit/s and guarantee quality of service (QoS).

We chose to adopt the IEEE 802.15.3 standard [3] for our system, because it supports QoS, provides various power management modes, security, ad hoc networking, and its physical layer supports data rates from 11 to 55 Mbit/s. For a better understanding of the presented work, a brief summary of the 802.15.3 medium access control (MAC) protocol is given in section 2. The standard specifies the MAC and physical (PHY) layer. In this paper, however, we will focus solely on the modeling of the MAC protocol using the specification and description language (SDL) [4].

SDL is an internationally standardized language for specifying and describing systems. It is a formal language, which means that it is possible to analyze and interpret SDL descriptions unambiguously. SDL tools, e.g. the Telelogic Tau SDL tool suite [5], allow models to be simulated and help to verify the modeled system. SDL has been widely used by telecommunications engineers and standards organizations for protocol specification, high-level system specification, prototyping, design, code generation, and testing. In section 3, the fundamental concepts of SDL are explained in order to lay the foundation for the main focus of this paper. A more detailed introduction to the application of SDL for wireless standards and protocol specification can be found in [6].

Modeling the high-level system behavior is often the first step in the design flow. In section 4, we will present our hardware/software co-design flow and how we will end up with a system implementation starting from the behavioral description. We will also compare our design flow to other approaches found in the literature.

The main part of this paper will be focused on our SDL model of the 802.15.3 MAC protocol. In particular, the object-oriented approach and layered architecture that led to a clear, maintainable design and a good starting point for the next steps in the design flow will be presented in section 5. Our results will be summarized at the end of paper.

1.1 Related Work

Many examples of successful adoption of SDL for communication protocol development have been described in the literature. In [7], the authors propose a formalized method to automatically derive an optimized parallel protocol implementation from an SDL specification. Other examples for embedded real-time communication protocol developments using SDL include work on a DECT implementation [8] and the TUTWLAN MAC protocol [9]. SDL has also been used to develop behavioral models of the IEEE 802.11 and HIPERLAN/2 MAC protocols [10].

Practical experiences from the development of SDL-based software for an embedded system and a comparison with a C-based approach are presented in [11].

SDL models for the 802.15.4 and 802.15.1 MAC protocols have been included in the standards as an informative annex to unambiguously specify the protocol behavior [12],[13].

TTPCom has announced an 802.15.3 MAC implementation on their web site [14], but does not provide any further information.

2 Overview of the IEEE 802.15.3 MAC Protocol

The 802.15.3 MAC protocol is specified in the 802.15.3 standard [3]. It provides for wireless connectivity between mobile stations and offers data rates from 11 to 55 Mbit/s. In 802.15.3, a network is named *piconet* and is formed by several devices (*DEVs*), as shown in Fig. 1. One DEV is required to assume the role of the piconet coordinator (*PNC*). The PNC provides the basic timing for the piconet through the

periodic emission of a beacon. Additionally, the PNC manages the QoS requirements, power save modes and access control to the piconet. Because 802.15.3 piconets form without pre-planning and for only as long as the piconet is needed, this type of operation is referred to as an ad hoc network [3].

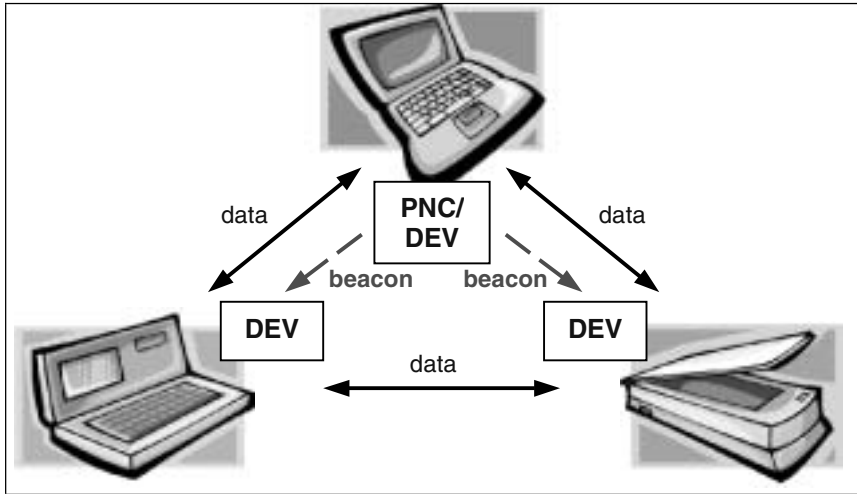


Fig. 1. 802.15.3 piconet elements [3]

A piconet is formed when a DEV that is capable of acting as the PNC begins transmitting beacons. One of the primary functions of the PNC is to transmit a beacon with appropriate information about the piconet. In order to participate in a piconet, a DEV needs to join the piconet using the association process described in the standard.

Timing in the 802.15.3 piconet is based on the *superframe*, which is illustrated in Fig. 2. The superframe is composed of three parts:

- The beacon, which is used to set the timing allocations and to communicate management information for the piconet.
- The contention access period (CAP), which is used to communicate commands and/or asynchronous data if it is present in the superframe.
- The channel time allocation period (CTAP), which is composed of channel time allocations (CTAs) including management CTAs (MCTAs). CTAs are used for commands, isochronous streams and asynchronous data connections.

The CAP uses CSMA/CA for the medium access. The CTAP, on the other hand, uses a standard TDMA protocol where the DEVs have specified time windows.

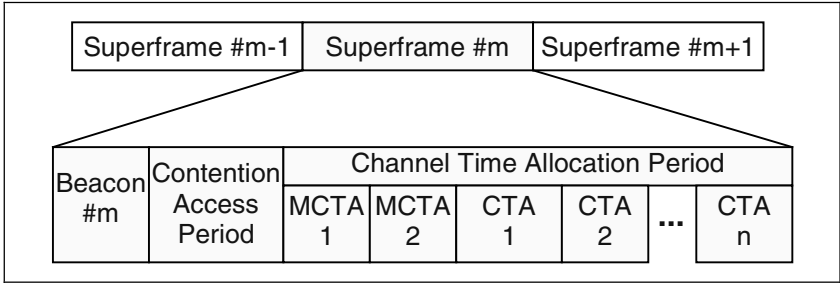


Fig. 2. 802.15.3 piconet superframe [3]

The MAC layer offers services to higher layer protocols and the device management entity via two service access points (SAPs), the MAC and MLME (MAC layer management entity) SAPs. The MAC SAP offers asynchronous and isochronous data transport services, whereas the MAC protocol is controlled by primitives sent through the MLME SAP.

3 Specification and Description Language (SDL)

Until recently, embedded systems designers used assembly languages as their primary software tools. However, with increasing complexity of the systems to be designed, there was a demand for higher-level description languages and sophisticated design tools. SDL can be considered as a programming language with a graphical user interface that offers high-abstraction level programming constructs to the designer. SDL models can be simulated, which allows the verification of the system functionality during an early stage of the design flow.

The system behavior in SDL is based on communicating *extended finite state machines* that are executed concurrently. State machines are represented by *SDL processes*. Processes communicate with each other and the system environment by exchanging *asynchronous signals* that may carry any number of parameters. SDL also provides *timers* that can be configured to generate signals at defined points in time. Each process in an SDL system contains a FIFO (First-In-First-Out) input buffer (with infinite space) into which the received signals are queued. Signal reception triggers a state transition [9].

SDL models are hierarchically structured. The system level, which is the top level, consists of blocks connected via channels. Channels are used as signal carriers. Each block may contain any number of sub-blocks. The lowest level sub-block contains the actual processes. A process may have local variables and may contain procedures. Processes communicate within the same or across different blocks via signal routes. Procedures are the lowest level in the functional hierarchy and have their own local scope.

4 Hardware/Software Co-design Flow

The design flow that is generally applied for the design of complex systems consists of four phases: the specification, exploration, synthesis, and test phases.

In the specification phase, a behavioral model of the system based on the functional requirements, and constraints, is created. This model is validated or verified by means of simulation or formal methods. Next, in the exploration phase, based on estimates or performance simulations, an optimized system architecture and a mapping of the functional model to this architecture are selected. The system architecture may consist of several components such as general-purpose processors, ASICs, or busses. During the final steps of the design flow, software parts need to be compiled for the target processor, and hardware is synthesized. Finally, the system is integrated and tested.

High-level specification languages, such as SDL, accelerate the design flow, as the system can be modeled on a high abstraction level. This facilitates the design of complex systems, leads to less design errors, and allows simulating and formally verifying the model. Ideally, the behavioral model that was created in the specification phase serves as the basis throughout the complete design flow. SDL tools allow the automatic generation of C code from the SDL model. This code can be compiled for the target processor to end up with a software implementation of the system.

However, there are still no mature hardware/software co-design tools that operate solely on the SDL model, but use the generated C code as the basis for simulations, profiling, and implementation. This code is far from being optimal, so that, in particular, data processing algorithms are slowed down. Furthermore, any parallelism that was expressed in the model by means of concurrent processes is lost when simulating sequential C code. Communication and scheduling overheads incurred by the SDL run-time system heavily influence performance simulations. Therefore, it is questionable how well the performance results reflect the real achievable performance, unless a lot of effort is spent on precisely estimating any avoidable overhead. In addition to that, the size of the compiled code is much larger than what could be achieved without using SDL, since the SDL run-time library has to be included. Typically, the compiled code amounts to about 500 kBytes.

SDL is therefore only used as a basis for thorough testing and extensive protocol simulations and not for the actual software implementation.

Other proposed design methodologies that follow the approach of continuously refining the behavioral model and using it for performance estimations, hardware/software partitioning, and an optimized implementation are, for example, the SystemC language [15] and Polis [16]. In our opinion, these approaches are still not mature enough, lack tool support, or do not provide required abstraction levels.

Our SDL model is designed in such a way that it can easily be used for a hardware/software system implementation. This is due to a clear separation of functional blocks as presented in the following section. However, an investigation on how to reliably estimate the performance of the system for different target architectures and hardware/software partitions is still an active research issue and is not supported by currently available design tools.

5 Behavioral SDL Model

In section 5.1, we will outline our design goals and basic ideas behind the design of the behavioral SDL model of the 802.15.3 MAC protocol. Next, the hierarchical structure as well as the functionality contained in the individual blocks will be described in sections 5.2 and 5.3, respectively.

5.1 Design Goals

Our primary goal for the design was to create an *abstract functional model* of the protocol described in the standard. We wanted to ensure correct protocol operation and that the behavior would be reflected by the model. By simulating the SDL model, we could verify its design.

Secondly, as we consider using the 802.15.3 MAC protocol as a basis for proprietary protocol extensions and modifications, it was required to design the model in such a way that it could be *easily extended or adapted*. The reference SDL models for the IEEE 802.15.4 [12] and 802.15.1 [13] standards published by the standardization body do not show this property. In fact, from our point of view, these models seem to be unstructured and hard to understand, as they lack documentation and the given names for signals, identifiers, states etc. are not self-explanatory.

A third, however, not so important design goal was to *ease the implementation* of the system. As outlined in the previous section, we do not follow the automated C code generation approach for system implementation. Instead, optimized software (C++, C, or assembler code) and hardware (VHDL code) is to be developed from scratch. Therefore, it was required that the model is based on concepts that are also available for the implementation. Additionally, a clear modularization and the decoupling of functionality eases hardware/software partitioning and implementation. This is best achieved by following an object-oriented design approach.

5.2 Model Architecture

Our top-level SDL system consists of a variable number of Station instances that are connected with each other through the Airlink block. Stimuli to the individual stations as well as station configuration parameters are generated from within the Testbench block. This is shown in Fig. 3. The interfaces of the Station block to higher protocol layers are the MAC, MLME, and PLME SAPs as defined in the standard. More details on the Station block structure are given below.

The Testbench contains SDL processes that send requests through the MAC and MLME SAPs of the Station block. This makes it possible to simulate the operation of a piconet consisting of several mobile devices. Configuration parameters are required, because it is necessary to distinguish between individual stations. We use direct addressing to send requests to the respective stations. It is possible to run a test program that is stored in a file instead of fixing the order of events in an SDL process.

In the Airlink block, a simple broadcast channel model is used. Frames that are transmitted by a station are received by all other stations at exactly the same time. It is

possible for frames to collide or to be received in error due to random bit error insertion. This is indicated to the receiving stations.

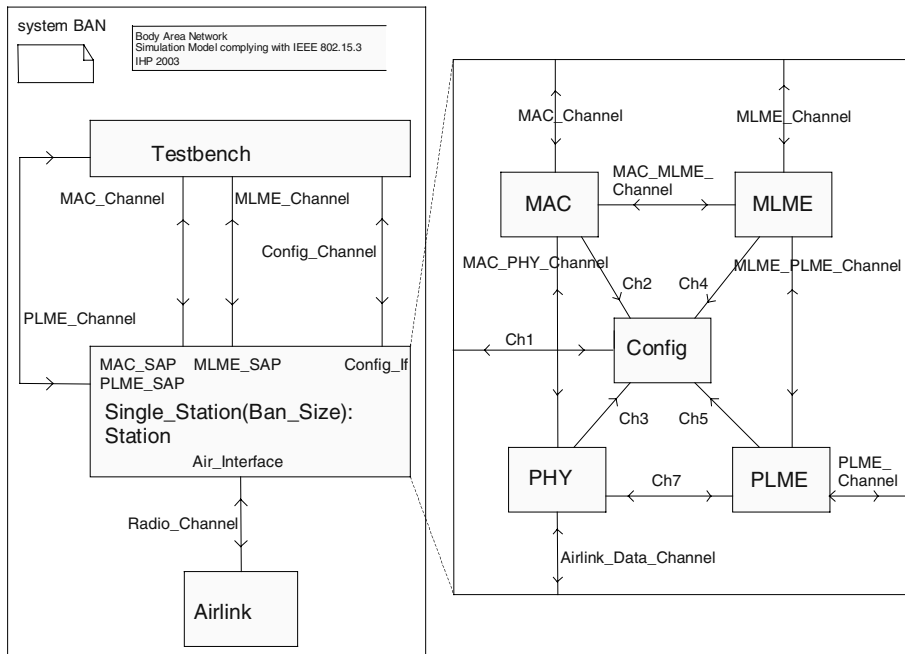


Fig. 3. Structure of the 802.15.3 MAC model

Referring to the 802.15.3 standard, the Station block contains five sub-blocks. These are MAC, MLME, PHY, PLME, and Config as shown in Fig. 3. Only the MAC and MLME block functionality – the focus of the next section – will be used for the final MAC layer implementation. The PHY and PLME sub-blocks contain abstract models for frame transmission, reception, and channel sensing as well as the PIB (personal information base) attributes of the PHY layer. The PHY layer functionality, such as synchronization, equalization, or coding, are outside the scope of this model. The Config block is only required for SDL-specific reasons in order to handle more than one station in the piconet.

5.3 Behavioral Description

The MAC layer functionality is contained completely in the MAC and MLME blocks, which corresponds to a clear separation of the so-called *data path* and *control path*. All data flow processing, such as

- CRC sum calculation,
- encryption and decryption of the frame payload,
- interfacing with the PHY layer, and
- frame buffering

is modeled in the MAC block. The complexity of the MAC block functionality is rather low and the modeling of the mentioned algorithms was straightforward, therefore we will focus on the control path in this paper. Furthermore, it is not required to optimize data path processing algorithms in the SDL model, since at this point in the design flow we were aiming at a functional model of the protocol. Algorithms will be optimized later for hardware/software partitioning and implementation.

The MLME block is responsible for controlling the operation of the MAC block, maintaining protocol operation, and handling requests received via the MLME SAP. The design of the MLME block was driven by an object-oriented approach, which means that we first tried to identify basic modular units that are responsible for a single task and provide an interface, but not implementation details, to its clients. These units are known as classes in the object-oriented domain and are represented by SDL processes. The exchange of signals between processes corresponds to the invocation of methods of an object. The concepts of concurrent process execution and asynchronous communication, which are native to SDL, have the great advantage of not anticipating any implementation choices regarding the hardware or software mapping and the communication between processes.

The modularization approach has got the advantage that many designers in a team can work on the SDL model in parallel. Additionally, it leads to a decoupling of the individual modules (processes), so that the model can be modified or extended easily, without breaking the system. In short, applying the object-oriented design methodology for protocol design introduces the same benefits as seen for the development of large software systems, i.e. reduced complexity, understandability, etc. Based on the model for the MLME block, our approach will be illustrated in some detail below.

5.3.1 Service Layers in the MLME Block

The SDL processes in the MLME block can be grouped into three conceptual service layers and one management plane, as shown in Fig. 4. Additionally, within each layer, we identified those processes that are only needed for a PNC-capable device. This layering approach and the separation of PNC-specific functionality further enhances the clarity of the model and gives initial indications for a potential hardware/software partitioning.

The lowest service layer in the MLME block is called *Transport Engine*. It provides to the upper service layers the ability to receive and transmit service data units (SDUs) such as commands, beacons or data units. This means that any upper-layer processes do not have to deal with the channel access procedure, fragmentation and reassembly, retransmission, exact timing of transmission and reception, and so on. The timers in this layer require an accuracy of 1 μ s.

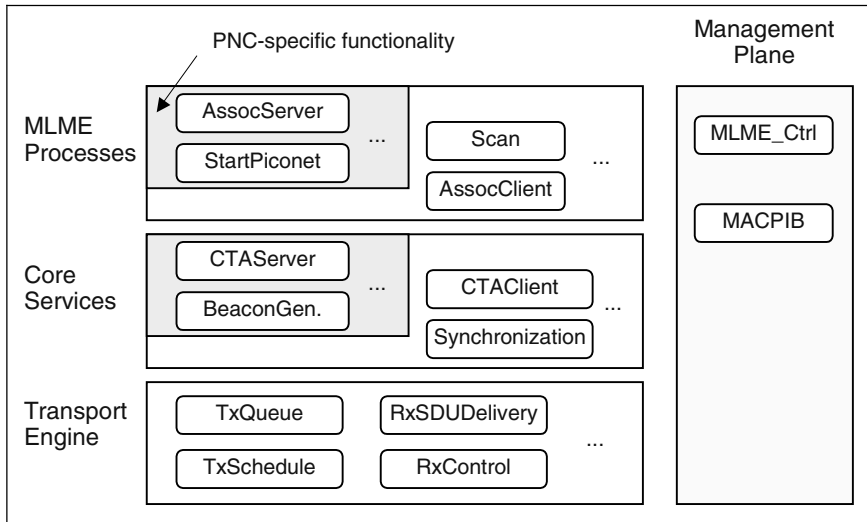


Fig. 4. Functional layering of the MLME block processes

On top of the Transport Engine, the *Core Services* are placed. The processes in this layer are responsible for maintaining the piconet operation. The CTAServer process, for instance, manages channel time allocations in the superframe. A scheduling algorithm determines which stations are granted channel access based on previous channel time reservations. The Synchronization process watches the reception of beacons and takes action if the beacon was lost in several consecutive superframes.

The highest service layer contains the so-called *MLME Processes*. These are, for example, the StartPiconet, Scan, AssocServer, or AssocClient processes. Their behavior is defined in the 802.15.3 standard. Note, that the Core Services and MLME Processes do not require timers with an accuracy of 1 μ s, but millisecond timers are sufficient.

Finally, the *Management Plane* contains the MLME_Ctrl process, which handles requests received via the MLME SAP and controls the overall station behavior, and the MACPIB process that manages the PIB attributes.

Our layered approach leads to a decoupling of the functional modules of our model. Additionally, it facilitates the introduction of non-standard protocol extensions or new frame types. If desired, any additional functionality that relies on the basic frame exchange mechanisms can be placed above the Transport Engine layer with no or little impact on the overall model. Likewise, the basic channel access scheme or interframe spaces can be adapted by modifying the model at a single well-defined place.

The presented structure of the MAC protocol functionality gives good indications about which functions should be implemented in hardware and which in software. This is due to the fact that all bulk data processing is located in the MAC block and all time-critical operations can be found in the Transport Engine layer.

5.3.2 TxQueue Process

As an example for a module in the Transport Engine layer, we will present the SDL process called TxQueue. It is the responsibility of this process to queue service data units, i.e. beacons, commands or data units, for later transmission on behalf of other processes. This is initiated by sending a TxAddBeacon.request, TxAddCmd.request or TxAddData.request. When the SDU has been transmitted successfully, has timed out or has reached the retransmission limit, this is indicated via the TxSDUStatus.indication signal to the respective client process and the SDU is removed from the queue.

The TxQueue process will fragment SDUs into several frames, if necessary. Only if all fragments have been transmitted successfully and in the correct order, the SDU is removed from the queue. Another SDL process, called TxSchedule, queries the TxQueue process for the next frame to be transmitted. This can be either a beacon frame, a frame that is to be sent during the Contention Access Period (CAP) or during a Channel Time Allocation (CTA) for that device. The TxQueue process, then, determines what frame is to be transmitted next, based on priorities, remaining lifetime etc. It also performs aging of the queues in regular intervals, so that SDUs are removed from the queue when their maximum queuing lifetime has expired. Note, however, that the TxQueue process is not responsible for maintaining the superframe timer and observing the channel access procedure. This is modeled in the TxSchedule process. The TxSchedule process also informs the TxQueue process about successful or failed frame transmission via the TransmissionStatus.indication signal.

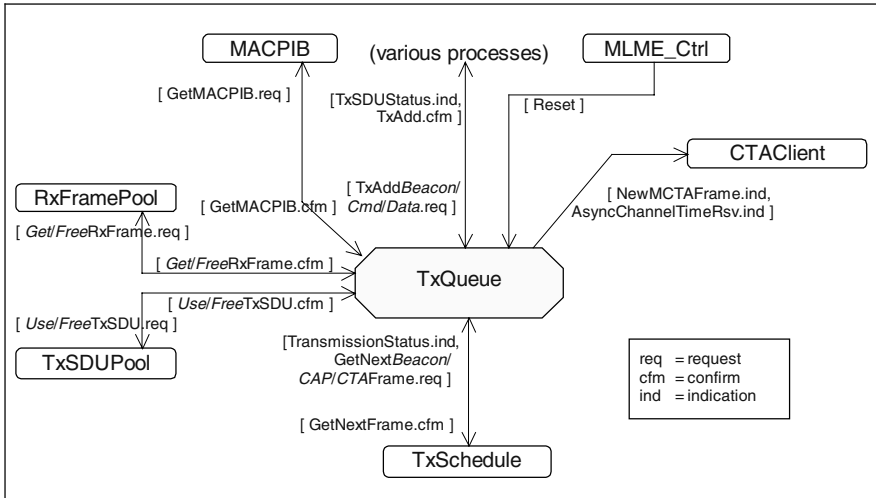


Fig. 5. Process interaction diagram for the TxQueue process

Additionally, the TxQueue process initiates a request for more channel time for frame transmission when a new SDU is queued and the current channel time reservation for that station is insufficient by sending NewMCTAFrame.indication and AsyncChannelTimeRsv.indication signals to the CTAClient process.

The evaluation of delayed acknowledgment frames also needs to be done in the TxQueue process. Therefore, the process is connected to the RxFramePool process in the MAC block. In Fig. 5, the relationships between the TxQueue process and other processes together with the signals that are exchanged between them are shown. A discussion of all the processes in the SDL model and their inter-relationships is outside the scope of this paper.

6 Conclusions

We have presented our SDL model for the 802.15.3 MAC layer. SDL is a system modeling language on a high abstraction level. The system behavior is expressed by concurrent processes that communicate asynchronously with each other. SDL processes can be described as extended finite state machines. We followed an object-oriented design approach for the development of the SDL model. Classes – the basic functional units in object-oriented design – are represented by SDL processes.

Each process is responsible for a single task and presents a well-defined interface to its clients. By applying the object-oriented methodology, we introduced many advantages of this methodology, for example, it facilitates teamwork, reduces complexity, leads to a clear, extensible, and re-usable design. Furthermore, we were able to structure the functionality of the model in three layers. This reduced the amount of coupling (interdependencies) between the processes and led to a model that is easily understandable. The SDL model can be used as the basis for our hardware/software co-design flow of a communication system.

References

- [1] IEEE Standard 802: Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, 1999.
- [2] Bluetooth SIG: Specification Of The Bluetooth System, 1999.
- [3] IEEE Standard 802: Part 15.3: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for High Rate Wireless Personal Area Networks (WPAN), 2003.
- [4] ITU-T: ITU-T Recommendation Z.100 (11/99). SDL: Specification and Description Language, 1999.
- [5] Telelogic AB: Telelogic Tau SDL Suite, 2003. Available from <http://www.telelogic.com/products/tau/sdl>
- [6] M. Graney: Speeding Up Wireless Standards Development. In: CommsDesign, 2000. Available from <http://www.commsdesign.com/main/2000/09/0009stand.htm>
- [7] S. Leue and Ph. Oechslin: From SDL Specifications to Optimized Parallel Protocol Implementations. In: M. Ito and G. Neufeld (eds.), *Workshop Proceedings of the Fourth International IFIP Workshop on Protocols for High Speed Networks*, pages 308–328, 1994.
- [8] C. Drosos, M. Zayadine, and D. Metafas: Embedded real-time communication protocol development using SDL for ARM microprocessor. In: *Dedicated Systems Magazine*, Q1 2001, pages 37–43, 2001.

- [9] M. Hännikäinen, J. Knuutila, T. Hämäläinen, and J. Saarinen: Using SDL for Implementing a Wireless Medium Access Control Protocol. In: *IEEE International Symposium on Multimedia Software Engineering (MSE 2000)*, pages 229–236, December 2000.
- [10] E. Grass, K. Tittelbach-Helmrich, U. Jagdhold, A. Troya, G. Lippert, O. Krüger, J. Lehmann, K. Maharatna, K. Dombrowski, N. Fiebig, R. Kraemer, and P. Mähönen: On the Single-Chip Implementation of a Hiperlan/2 and IEEE 802.11a Capable Modem. In: *IEEE Personal Communications*, December 2001.
- [11] S. Bläsius, S. Karg, J. Maier, and G. Kohler: Development of SDL-based software for an embedded system - practical experiences. In: *2nd Workshop on SDL and MSC (SAM2000)*, Grenoble, 2000.
- [12] IEEE Standard 802: Annex D: Formal description of the 802.15.4 operation. In: *Draft Standard for Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Low Rate Wireless Personal Area Networks (LR-WPANs)*. February 2003.
- [13] IEEE Standard 802: Annex B: Formal description of the 802.15.1 operation. In: *Part 15.1: Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications for Wireless Personal Area Networks (WPANs™)*. 2001.
- [14] TTP Communications plc: 802.15.3 MAC, July 2003. Available from: http://www.ttpcom.com/ttpcom/products/802_15_3_MAC.html
- [15] Open SystemC Initiative: <http://www.systemc.org>
- [16] Polis homepage at the University of California at Berkeley: <http://www-cad.eecs.berkeley.edu/~polis/>

Performance of Phase Modulated Systems Using Fully Saturated Power Amplifiers

Qi Lu and Qingchong Liu

Electrical & Systems Engineering Department
Oakland University
Rochester, MI 48326, USA
qlu@oakland.edu

Abstract. This paper studies the performances of BPSK and QPSK modulations amplified by fully saturated power amplifiers. To achieve better performance, both convolutional code and Turbo code are used. Viterbi decoding and iterative turbo decoding are employed in the receiver, respectively. Simulation results show that good BER performance can be achieved when signal is amplified by fully saturated power amplifiers and transmitted through AWGN channel. The SNR degradation is 0.7 dB for BPSK and 0.3-0.4 dB for QPSK at $\text{BER} = 10^{-5}$.

1 Introduction

Broadband wireless communications are attracting more and more attention. A large number of user terminals should be supported in those broadband wireless networks. As the data rate is much higher than that in the existing narrowband wireless networks, the user terminals need a much higher power level. Due to the quantity and the power consumption of user terminals, power amplifier cost and DC-to-AC conversion efficiency, which are unsolved, are key problems in broadband wireless communications. At present, most power amplifiers used in traditional narrow band wireless communication systems are employing class A output stages. The DC-to-AC conversion efficiency η is less than 30%. Linear power amplifiers are also too hard to be designed and manufactured for broadband wireless communication systems. Working with the bandwidth of hundreds of megahertz, the nonlinearity is unavoidable. It seems the best way is to implement power amplifiers working in fully saturated region. For example, the class F power amplifier, which works in the strongly saturated region, can achieve $\eta = 91\%$ [1]. Amplifiers working in nonlinear region cause severe distortion to the transmitted signal.

Some linearization techniques are helpful in RF design to lower the overall nonlinearity without significantly degrading the efficiency. Because they generally complicate design, require various adjustment, and come less effective as device characteristics change with temperature and power, such techniques are not implemented in low-cost portable terminals. [2]

Predistortion method tries to combat the nonlinearity of RF amplifier not during the process of power amplifier design, but in the modulation before the

power amplifier. It adds inverse filters in front of power amplifiers so that the combination response of the inverse filters and the power amplifiers can have less nonlinearity. Predistortion has been widely studied. In [3], by using predistortion, 1.4 dB was achieved at $\text{BER} = 10^{-5}$, the degradation from linear curve is only 1.0 dB.

Although predistortion improves the performance, it increases the system overall cost. When a fully saturated power amplifier is implemented in the transmitter, we cannot take the advantage of predistortion. In a communication system employing a fully saturated power amplifier, the output signal of the power amplifier is either a constant positive value or a constant negative value, so that the effect of the inverse filter is totally removed from the transmitted radio. Predistortion is impossible to be implemented before the fully saturated power amplifier to remove the nonlinearity caused by the full saturation of the power amplifier.

In this paper, the performances of BPSK and QPSK amplified by fully saturated power amplifiers are presented. Another objective of this paper is to reveal if some powerful coding schemes used in linear channel are still suitable for this fully saturated distortion channel. Convolutional code and turbo code are applied in the transmitter, and Viterbi decoding and iterative decoding are implemented in the receiver, respectively.

2 System Model

The block diagrams of the communication systems studied in this paper are shown in Fig. 1 and Fig. 2.

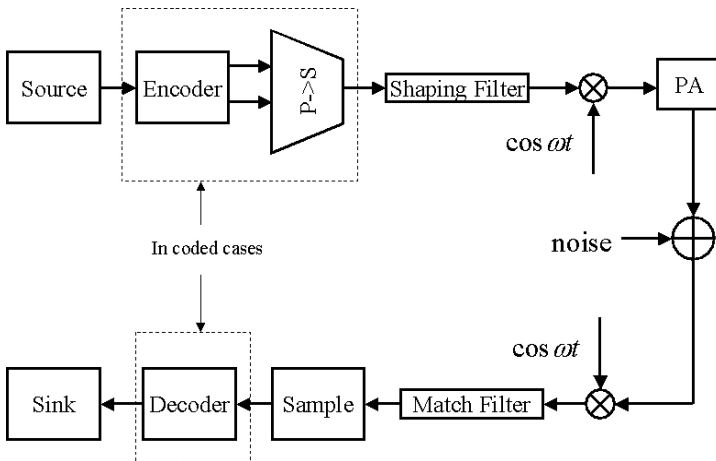


Fig. 1. Diagram for BPSK and Coded BPSK.

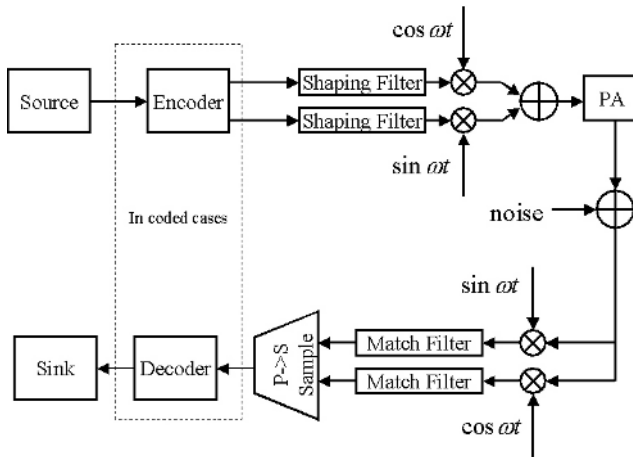


Fig. 2. Diagram for Coded QPSK.

Fig. 1 shows the block diagram for a communication systems employing BPSK. The data sequence is generated by a binary symmetric random number generator. In coded cases, it is fed into the encoder (convolutional or turbo). Since the output of the encoder is multibit, the parallel output of the encoder is serialized before a shaping filter. After being multiplied with carrier and amplified by a fully saturated power amplifier, the modulated signal is sent through a AWGN channel. In uncoded case, the data sequence from the generator is filtered and multiplied with the carrier directly, then passes the fully saturated power amplifier. In the receiver, down-converted signal plus noise is filtered by a matched filter and sampled. The sampled sequence is fed into the decoder (soft viterbi or BCJR iteration). The output of the decoder is compared with the original data to get the BER.

Fig. 2 is the diagram for QPSK. Coded or uncoded data are mapped into in-phase and quadrature signals. They are filtered separately. After being multiplied with quadrature carriers, the sum of in-phase and quadrature signals is amplified by the fully saturated power amplifier.

Both the convolutional code and the turbo code are employed. The convolutional code is of rate $R = 1/2$, constraint length $K = 7$ and polynomials 133 and 171. For turbo code, we used the same encoding scheme as in CDMA2000 [4]. The polynomials are 13 and 15, the constraint length is $K = 4$ and the coding rate is $R = 1/2$. The encoder employs two systematic, recursive, convolutional encoders connected in parallel, with an interleaver which is in front of the the second recursive encoder. The outputs of encoders are punctured to achieve the designated data rate. During encoding, a tail sequence is added to make sure that at the end of each frame these two convolutional encoders go back to all-zero state. If the total number of information bits is N , the output include $2N$ encoded data symbols followed by 12 tail output symbols. For the rate $R = 1/2$, the puncture patterns for coded information bits and tail bits are shown in table 1.

Table 1. Puncture pattern for turbo code with $R = 1/2$ in CDMA2000.

Information Sequence		Tail Sequence	
Output	Pattern	Output	Pattern
X	11	X	111 000
Y_0	10	Y_0	111 000
X'	00	X'	000 111
Y'_0	01	Y'_0	000 111

In the tables above, a '0' means that the symbol will be deleted and a '1' means that the symbol will be passed.

The interleaver in the turbo encoder performs block interleaving of the data. It plays an important role in the encoder. The interleaving algorithm is described below:

- Step 1.** Determine the turbo interleaver parameter n where n is the smallest integer such that $N_t \leq 2^{n+5}$, and N_t is the frame length of the input;
- Step 2.** Initialize an $(n+5)$ -bit counter to 0;
- Step 3.** Extract the n most significant bits (MSBs) from the counter and add one to form a new value. Then discard all except the m least significant bits (LSBs) of this value;
- Step 4.** Obtain the n -bit output of a lookup table;
- Step 5.** Multiply the value obtained in Step 3 and Step 4 and discard all except the n LSBs;
- Step 6.** Bit-reverse the five LSBs of the counter;
- Step 7.** Form a tentative output address that has its MSBs equal to the value obtained in Step 6 and its LSBs equal to the value obtained in Step 5;
- Step 8.** Accept the tentative output address as an output address if it is less than N_t ; Otherwise discard it;
- Step 9.** Increase the counter and repeat Step 3 through Step 8 until all N_t interleaver output addresses are obtained.

The modified BCJR algorithm is implemented in the decoder [5]. The performance is simulated using one iteration and three iterations.

For the shaping filter and the match filter, we use the root raised cosine FIR filter. The sampling rate is 16 and the duration is 8 symbols. Filters with different roll-off factor β have different effects on the performance of the nonlinear system [6].

Notice that although there is no interference at the sampling point for all filters, different roll-off factor has different effect on the power distribution in time domain.

3 Simulation Results

For BPSK, suppose the input signal of the shaping filter is

$$s_i(t) = \sum_{i=-\infty}^{\infty} a_i \delta(t - iT_b)$$

where $a_i \in \{+1, -1\}$ and $\delta(t)$ is the Dirac impulse function. Hence the output of the shaping filter can be written as

$$s'_i(t) = \sum_{i=-\infty}^{\infty} a_i h(t - iT_b)$$

where $h(t)$ is the impulse response of the shaping filter. There is a low pass filter following the power amplifier in practice, so the harmonics are ignored in the transmitted signal. Hence transmitted signal can be written as

$$s_o(t) = \text{sign}(s'_i(t)) \cos 2\pi f_c t$$

where f_c is the carrier frequency. Figure 3 is the figure of BER without any coding scheme.

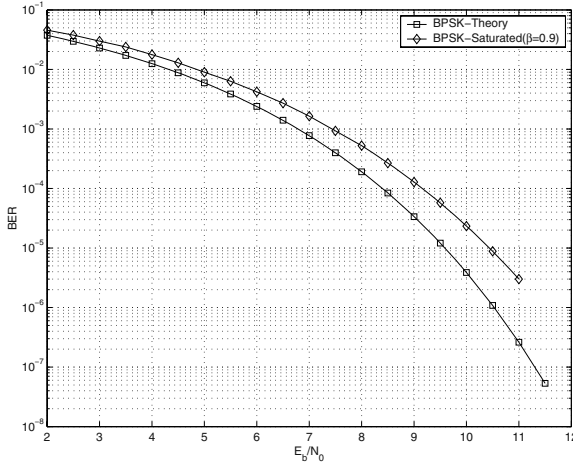


Fig. 3. Simulation results for uncoded BPSK with saturated power amplifier.

For coded BPSK, we employed convolutional code and turbo code. We can see there is little difference between $\beta = 1.0$ and $\beta = 0.9$ and the BER performance is improved. Figure 4 is the results for BPSK using convolutional code.

The turbo decoding algorithm used in this paper needs the estimation of E_b/N_0 and the distribution caused by the power amplifier makes the estimation

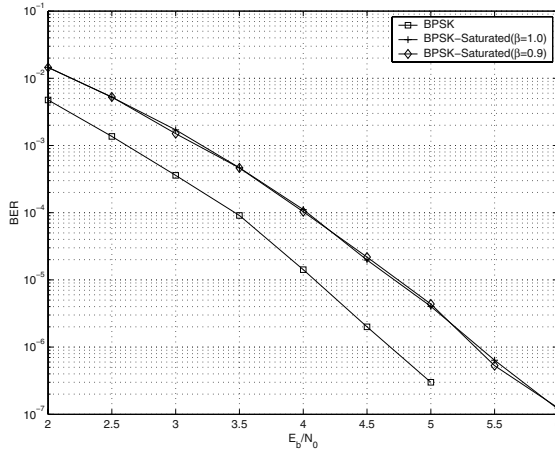


Fig. 4. Simulation results for coded BPSK (convolutional code, $R=1/2$, $K=7$, [133,171]) amplified by saturated power amplifiers.

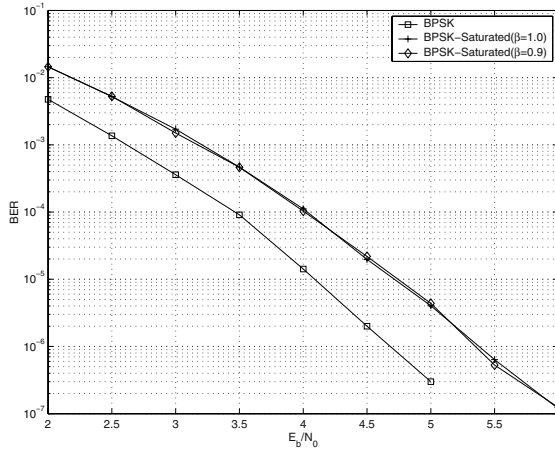


Fig. 5. Simulation results for coded BPSK (turbo code, $R=1/2$, $K=4$, [13,15], 3 iterations) amplified by saturated power amplifiers.

inaccurate. This may lead to performance degradation of the decoder in the studied nonlinear channel. The results for turbo code are shown in Fig. 5.

Because in-phase baseband signal and quadrature baseband signal are modulated with orthogonal carriers for QPSK modulation, the performance should be the same in the linear case. But for the saturated case, the modulated in-phase signal and quadrature signal interfere with each other when passing the fully saturated power amplifier. This makes the BER performance differ from that of fully saturated BPSK. If we omit the higher harmonics, the transmitted QPSK signal can be written as

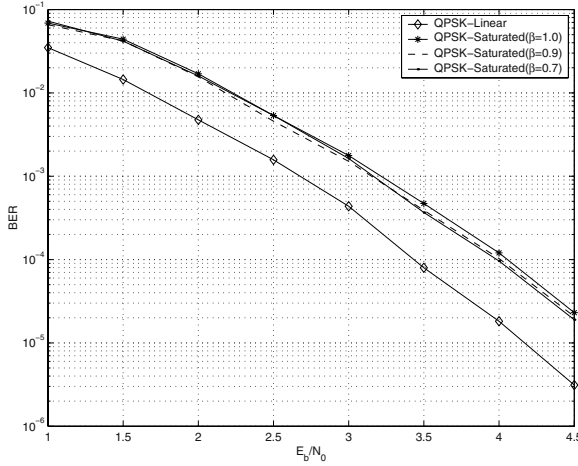


Fig. 6. Simulation results for coded QPSK (convolutional code, $R=1/2$, $K=7$, [133,171]) amplified by saturated power amplifiers.

$$s_o(t) = \frac{K s_I(t)}{\sqrt{s_I^2(t) + s_Q^2(t)}} \cos 2\pi f_c t - \frac{K s_Q(t)}{\sqrt{s_I^2(t) + s_Q^2(t)}} \sin 2\pi f_c t$$

where $s_I(t)$ is the in-phase baseband signal, $s_Q(t)$ is the quadrature baseband signal and K is a constant determined by the power amplifier.

In this paper, uncoded QPSK is not simulated. For Coded QPSK using Convolutional encode and Viterbi decoder, the same constraint length and polynomials as BPSK are used. Figure 6 is the simulation results for convolutional coded QPSK.

To take the advantage of iterative decoder, we compared two cases of turbo code : one has 1 iteration, the other has three. When 3-iteration decoder is implemented, the performance is much better than that of the convolutional code. Figure 7 and Fig. 8 show the results for turbo code QPSK simulation.

The results show that both convolutional code and turbo code can be used in this nonlinear system to improve BER performance.

4 Conclusions

In this paper, we study the performances of BPSK and QPSK transmitted by fully saturated power amplifiers. To improve the performance, different coding-decoding schemes are applied. Although a degradation follows, BPSK and QPSK with fully saturated power amplifiers achieve better performances than that reported in [3]. At $\text{BER}=10^{-5}$, there is only 0.7 dB degradation for BPSK. When QPSK is employed, the BER degradation decreases to 0.3-0.5 dB. It is also shown that convolutional code and turbo code can be used to achieve a better performance in this AWGN plus distortion noise channel.

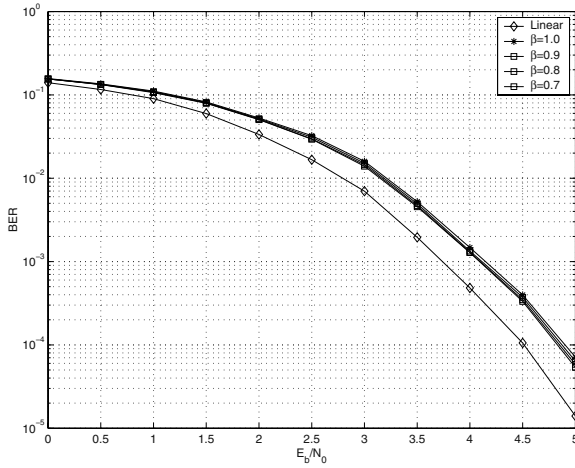


Fig. 7. Simulation results for coded QPSK (turbo code, $R=1/2$, $K=4$, [13,15], 1 iterations) amplified by saturated power amplifiers.

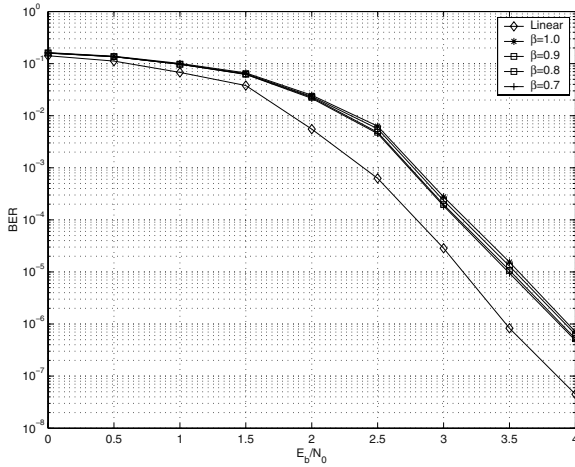


Fig. 8. Simulation results for coded QPSK (turbo code, $R=1/2$, $K=4$, [13,15], 3 iterations) amplified by saturated power amplifiers.

References

1. S. R. Cripps, *RF Power Amplifier for Wireless Communications*. Boston: Artech House, 1999.
2. B. RAZAVI, *RF Microelectronics*. New York: Prentice-Hall, 1998.
3. S. Vaughn and R. Sorace, "Demonstration of the TDRS Ka-band transponder," *Proc. 2000 IEEE Military Commun. Conf.*, vol. 2, pp 1055–1065, 22–25 Oct., 2000, Los Angeles, CA.

4. *Standards for CDMA2000 Spread Spectrum Systems*, EIA/TIA IS-2000.1–6.
5. C. Berrou, A. Glavieux and P. Thitimajshima, “Near Shannon limit error-correcting coding and decoding: turbo-codes,” *Proc. of ICC 1993*, pp. 1064–1070.
6. Q. Liu and J. Li, “Quasi-constant envelope OQPSK through nonlinear radio and AWGN channel,” *Proc. 2002 IEEE Military Commun. Conf.*, Vol. 1, pp 715–720, Oct. 2002, Anaheim, CA.
7. L. R. Bahl, J. Cocke, F. Jelinek and J. Raviv, “Optimal decoding of linear codes for minimizing symbol error rate,” *IEEE Trans. Inform. Theory.*, vol. 20, pp. 284–287, March 1974.
8. J. A. Heller and I. M. Jacobs, “Viterbi decoding for satellite and space communication,” *IEEE Trans. Commun. Technol.*, vol.19, pp. 835–848, October 1971.
9. L. Lee, A.R.Hammons,Jr. F. Sun and M. Eroz, “Application and standardization of turbo codes in third-generation high-speed wireless data service,” *IEEE Trans. Veh. Technol.*, vol.49, pp. 2198–2207, Sept. 2000.
10. A. J. Viterbi, “Convolutional codes and their performance in communication systems,” *IEEE Trans. Commun. Technol.*, vol.19, pp. 751–772, October 1971.
11. J. G. Proakis, *Digital Communications*. New York: McGraw-Hill, 1995.

On Connectivity in Ad Hoc Networks under Jamming Using Directional Antennas and Mobility

Guevara Noubir

College of Computer and Information Science
Northeastern University
Boston, MA 02118, USA

noubir@ccs.neu.edu

Abstract. In this paper we investigate the problem of maintaining connectivity under jamming in multihop ad hoc wireless networks. Connectivity is measured using a connectivity index, which indicates the probability that there exists a path between two nodes. We first show that connectivity can be drastically reduced with a relatively small number of jammers. We show that using sectorized antennas can maintain connectivity in the presence of a significantly higher number of jammers at the expense of higher average number of hops. Finally, we show that mobility allows further resiliency to jamming.

1 Introduction

Wireless communication is exposed to various denials of service attacks at all protocol layers. Robustness of wireless multihop ad hoc networks is essential to various applications both in the military context and in future commercial applications. Jamming is the most traditional technique to prevent wireless communication. Jamming can be malicious, aiming at preventing wireless communication in an area, but can also be due to non-interoperability of wireless standards. In a military context adversaries intentionally jam the communication channel to prevent nodes from correctly receiving data packets. In [1] we have shown that for a single hop omni-directional communication of *data packets*, an adversary can easily break the wireless link at a very low energy cost. Jammers do not need to be large high power transmission devices, they can be composed of a set of small low cost “cyber-mines” randomly spread over the area of jamming interest to the adversary. Non-malicious jamming can occur in both military and commercial communication. For example the 2.4GHz Industrial, Scientific and Medical band is crowded by multiple non-interoperable standards (e.g., IEEE802.11, Bluetooth, cordless phones in the US) and the 5GHz frequency band will also be used by various non-compatible standards (e.g., IEEE802.11, Hiperlan II). The presence of non-compatible communication leads to interference that can have the same effect as jamming. Multihop ad hoc networks have the advantage of being able to use multiple paths to maintain connectivity. In this paper, we show that, under jamming, the connectivity can drop drastically. However, the use of sectorized antennas can significantly improve connectivity. In the past,

sectorized antennas were only used at fixed base-stations of access points. But the advent of compact sectorized antennas will make their integration feasible for mobile devices. For example, Antenova [2] already sells 5 and 16-sectorized antennas of small dimensions (5cm x 15cm). Some new PDAs such as the Wanda from Texas Instruments [3] already integrate 4 antennas to allow co-existence of Bluetooth, IEEE802.11, and GPRS (General Packet Radio Service). This integration trend will continue specially for high frequencies bands because the antenna size is usually directly related to the signal wavelength. Directionality will also be provided using smart antennas' beam forming techniques or MIMO (Multiple Input Multiple Output) technology.

Connectivity of ad hoc networks has been extensively studied and various results were obtained [4-7]. Previous research has mainly addressed the problem of determining the optimal transmission range or nodes degree to maintain connectivity. More recent studies used percolation theory [8-10] both in a 0-1 connectivity model and in interference based connectivity model. Analysis based on percolation theory proved to be a powerful tool in exhibiting phase transition behavior for connectivity. In [11], it was shown that when the number of nodes goes to infinity the minimum range for achieving k -connectivity (existence of k disjoint paths) is the same as the minimum range for each node having k neighbors. Some recent studies addressed the modeling issues related to links failures [12] by looking at the graph minimum cut. In addition to physical layer jamming several DoS techniques can be applied at higher protocol layers of systems such as IEEE802.11 (e.g., by forcing the backoff window to remain at its maximum) or Bluetooth MAC (e.g., by destroying some control packet), routing (e.g., by injecting erroneous or destroying control routing packets), and transport protocols (e.g., by forcing TCP multiplicative decrease to keep the congestion window small) [13-20]. However to the best of our knowledge no work has been done in investigating the use of directional antennas and mobility in maintaining connectivity.

In Section 2, we present the problem of jamming in multihop ad hoc networks, describe traditional jamming and anti-jamming techniques, and introduce the connectivity measures that we will be using to evaluate our techniques. In Section 3, we analyze the network connectivity under jamming when using omni-directional antennas. In Section 4, we show how sectorized antennas can increase the tolerance of the network connectivity to a higher number of jammers. In Section 5, we show how mobility can further increase the network connectivity.

Notation:

N : the number of communicating nodes

R : nodes communication range

NJ : the number of jamming nodes

JR : range of jamming

2 Problem of Jamming and Connectivity in Ad Hoc Networks

2.1 Jamming Principles

Traditional jamming techniques address the energy cost of jamming a single symbol/bit in a communication. The jamming capability of a single symbol is a function of the jammer power, the transmitter power, the antennas gains (from jammer to receiver, receiver to jammer, transmitter to receiver, and receiver to transmitter), the communication receiver bandwidth, the jamming transmitter bandwidth, the range between the transmitter and receiver, the range between the jammer and receiver, the jammer signal loss, and the communication signal loss [21]. Classical jamming consists in injecting an interfering signal that submerges the signal at the receiver. Several interfering waveforms can be used such as noise modulated FM, noise bursts, or continuous wave (CW) tone. The jammer can also play-back a previously recorded signal. Resistance to jamming is traditionally achieved by tuning various parameters such as transmission power, directional antennas, and receiver communication bandwidth. In the next paragraph, we describe one of the most common and efficient bit-level anti-jamming techniques; namely *spread spectrum*.

Traditionally, jamming strength is measured through the jamming-to-signal ratio defined as follows [21]:

$$\frac{J}{S} = \frac{P_j G_{jr} G_{rj} R_{tr}^2 L_r B_r}{P_t G_{tr} G_{rt} R_{jr}^2 L_j B_j}$$

P_j : jammer power

G_{jr} : antenna gain from jammer to receiver

G_{rj} : antenna gain from receiver to jammer

R_{tr} : distance from transmitter to receiver

L_r : communication signal loss

B_r : communications receiver bandwidth

P_t : transmitter power

G_{tr} : antenna gain from transmitter to receiver

G_{rt} : antenna gain from receiver to transmitter

R_{jr} : distance from jammer to receiver

L_j : jammer signal loss

B_j : jamming transmitter bandwidth

Protection against jamming in wireless communication is usually achieved by reducing the jamming to signal ratio. The obvious technique to reduce the jamming-to-signal ratio is based on increasing the transmission power level. However, this technique is not very efficient and is usually used as a last solution. The most commonly used anti-jamming technique is *spread spectrum* [22]; it relies on reducing B_r/B_j . These techniques force the jammer to spend much more energy than the sender. This is achieved by forcing the adversary to jam over a larger frequency bandwidth than the effective receiver/communication bandwidth. The typical value of the spread spectrum processing gain in military communication is between 20 dB and 30 dB. Spread spectrum technology uses a pseudorandom sequence to spread a signal over a much larger frequency band than what is required for its transmission. Correlating the received signal with the pseudorandom sequence carries out the despreading operation. There are two main spread spectrum techniques, namely: the direct sequence

technique and frequency hopping. If the pseudorandom sequence is unknown to the jammer, then the spreading operation achieves a processing gain G in the signal-to-jamming ratio. To successfully jam a communication the adversary would have to compensate this processing gain by increasing its transmission power. As will be explained in the next paragraph, spread spectrum is not sufficient to fully protect jammers. In this paper we consider the reduction of the antenna gain from the jammer to the receiver and use it in the context of multihop communication.

One has to note that reducing the jamming-to-signal ratio does not necessarily lead to complete resiliency to jamming. This is due to other vulnerabilities introduced by higher protocol layers. In [1], we have shown that it is easy to jam existing wireless data networks at a very low energy cost. We have shown that the jamming cost for IP over IEEE802.11 can be as low as 10^{-4} the cost of the communication. This is done by destroying a chosen very small fraction of the data packet to make the CRC wrong. An adversary can therefore deploy a set of low-cost *cyber-mines* that can passively detect packets (or some packets such as routing control packets) and destroy them. These cyber-mines can therefore last for long period of times. They not only can prevent communication but also might be used to force communication through paths where more powerful nodes intercept the traffic. We have shown that using a cryptographic interleaver with error correction codes can reduce the problem of jamming into resiliency to noise over a binary symmetric channel (BSC). Therefore, the capacity bound under jamming is given by Shannon's theorem as the channel entropy. The achieved result allows providing much better resiliency to jamming specially compared to existing WLAN standards such as IEEE802.11 and bluetooth.

In this paper, we will investigate the impact of the antenna gain factors to reduce the jamming-to-signal ratio and its impact on connectivity in multihop ad hoc networks. We propose to reduce the jammer to receiver antenna gain factor in the jamming to signal ratio. A simple approach to achieve this is by using sectored antennas, which results in isolating the jammers. The limitation of this technique is that complete isolation is not possible for example when the jammer and transmitter happen to be on the same sector of the receiver. Therefore, the use of multipath is helpful because there might be another receiver that cannot be jammed for the same transmitter. In practice the jamming range can be bigger or smaller than the communication range depending on the transmitter's power, jammer's power, spreading factor, etc. In our simulation we make the assumption that the jamming range is equal to communication range. Our results can be extended to the power controlled jamming range case. Our connectivity analysis considers an arbitrary jamming and communication range. However, a more careful analysis has to be applied to investigate energy efficient jamming/anti-jamming strategies.

2.2 Connectivity

A graph is said to be connected if there exists at least one path between any two nodes. Since jamming results in a directed graph, characterizing the level of connectivity of such a graph that is not connected is more difficult. Intuitively we character-

ize the level of connectivity of a graph by the average number of nodes that can be reached from any node of the graph. This can be measured using a function similar to the gamma index of the transitive closure of the connectivity graph. We first define a link that is not jammed. Then we define the connectivity index.

Definition 1: Let R be the communication range of the nodes, JS be the set of jammers, and JR be the jammer range. A link from node A to node B is said to be non-jammed if and only if:

$$d(A, B) < R \wedge \forall J \in JS : d(J, B) > JR$$

$d(A, B)$ denotes the Euclidian distance between the locations of node A and node B . We will later generalize this definition to the case of directional antennas. It is worth noting that links are not symmetric.

Definition 2: Let $G = (V, E)$ be the directed connectivity graph of a multihop ad hoc network after removing jammed links. Let $G' = (V, E')$ be the transitive closure of G .

The connectivity index of G is defined as: $\frac{|E'|}{|V|^2}$.

A connected graph has connectivity index 1, since its transitive closure is a clique. A graph partitioned into two equal size connected graphs has a connectivity index of 0.5. Therefore the connectivity index drops quickly with partitions and from a practical perspective maintaining a connectivity of 0.9 can be seen as a good result.

3 Jamming Omni-directional Communication

In this section we assume that the nodes communicate using omni-directional antennas. Therefore, a node will not be able to receive any data if it is within range of at least one jammer. However, this does not necessarily prevent the jammed node from transmitting to a non-jammed node, therefore creating an asymmetric network. We consider two cases, first when a large number of jammers are randomly spread over a large area A . In the second case we investigate the minimum number of jammers required to jam the whole area A .

3.1 Randomly Located Jammers

Let us assume that both the communicating nodes and the jammers are randomly distributed over a large area $A \gg \pi R^2$. The nodes are distributed according to a homogeneous spatial Poisson process of density (intensity) λ (which corresponds to N/A). Similarly the jamming nodes distribution intensity is μ (NJ/A). Such a scenario corresponds to a completely unplanned dissemination of jammers and communicating nodes. This also implies that if there were N nodes in any given region, their location would be independently, uniformly distributed over the region.

Proposition 1: The probability that the network is disconnected is lower bounded by the following formula: $1 - (1 - e^{-\lambda\pi R^2})^N e^{-\mu\pi J R^2 N}$.

Proof:

$$\begin{aligned} \Pr[\text{G is disconnected}] &\geq 1 - \Pr[\text{node is connected}]^N \\ &\geq 1 - \Pr[\text{other node within range} \wedge \text{no jammer within range}]^N \\ &\geq 1 - (1 - e^{-\lambda\pi R^2})^N e^{-\mu\pi J R^2 N} \end{aligned}$$

For the case of *non-jammed* communication, Penrose [11] has shown that in the limit when N grows to infinity, the range that leads to a connected network is the same as the range that leads to each node having one neighbor. The result is in fact more general and applies k -connectivity in geometric graphs. It is not obvious how this result can be generalized to the case of a geometric graph where some links are failing due the presence of a jammer. If this result could be extended to the presence of jammers, the proposition would provide a tight bound.

Another connectivity metric that one might consider is the probability of all the nodes being disconnected.

Proposition 2: The probability that all nodes are disconnected is given by: $[1 - e^{-\mu\pi J R^2} (1 - e^{-\lambda\pi R^2})]^N$.

Proof: A node is disconnected if it does not have any neighbouring node within its communication range or if there exists a jammer within its jamming range. Since the location of jammers is independent of location of the node, the probability of a node being disconnected is: $1 - e^{-\mu\pi J R^2} (1 - e^{-\lambda\pi R^2})$. Assuming that the nodes are independently disconnected leads to the proof.

The probability that the network becomes disconnected is very high in the presence of even few jammers. This probability quickly increases to one as a function of the number of nodes. The reason is that there will necessarily be some communicating nodes within range of a jammer. Therefore the connectivity index we have introduced is a better measure, because it gives the probability that any two nodes can communicate. Fig. 1, shows the increase in the probability of all nodes being disconnected (from Proposition 2) as a function of the number of jammers, for $N=400$ in an area of 2000×2000 , and with a communication/jamming range of 200. The figure indicates that all the nodes are with very high probability disconnected when the number of jammers exceeds 100. It also indicates a relatively sharp phase transition around 50 jamming nodes. This is confirmed by the simulation from Fig. 4, which shows that the connectivity index drops quickly as a function of the number of jammers (consider only the curve for 1-sectored antennas). This result is not surprising because it is expected that a large number of jammers (e.g., 100) would lead a completely communi-

cation denied area. Our goal is a better characterizing of the connectivity under jamming using the connectivity index we have defined earlier.

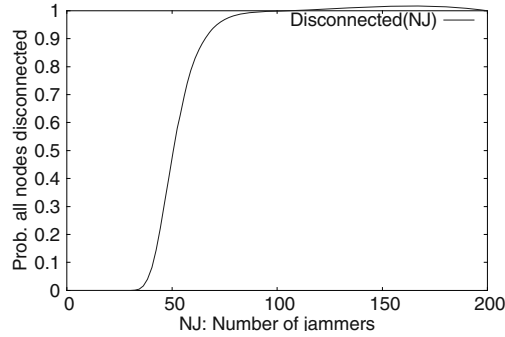


Fig. 1. Probability that all nodes are disconnected as a function of the number of jammers

3.2 Optimally Placed Jammers

If the adversary can choose the location of the jammers than the minimum number of jammers is given by the following theorem and their location is shown in Fig. 5. This is in fact similar to the problem of area coverage in cellular systems.

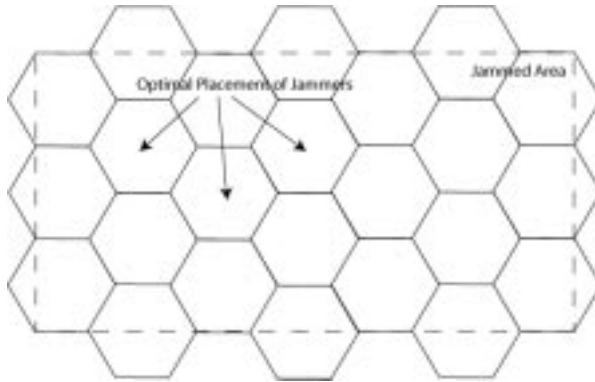


Fig. 2. Against omnidirectional communication, the optimal placement of antennas is at the centre of hexagonal cells

Theorem 1: Given a jamming range JR , the minimum number of jammers to cover an area A is (on the limit) $\frac{2A}{3\sqrt{3}JR^2}$.

Proof: This is a direct result from [23], where it is proven that the best covering of a plane with congruent circles is obtained with the hexagonal lattice covering.

In the case of an area of 2000×2000 , Theorem 1 indicates that 39 jammers can prevent all communication when the jamming range is 200. This is to be contrasted with the simulation results for randomly located jammers. One first conclusion is that not being able to control the location of the jammers drastically limits the adversary capability to prevent communication.

4 Maintain Connectivity Using Sectorized Antennas

In this section, we show that the use of sectorized antennas can provide significant resiliency to jammers. A sectorized antenna is a set of directional antennas that can cover all directions but can isolate the sectors. Usually sectorized antennas are used to improve energy efficiency by only radiating in the sector where the receiver is located. They also reduce the network interference level because they do not radiate on unnecessary sectors. We take advantage of the symmetry property of antennas. A sectorized antenna can also be used to receive on a single sector therefore ignoring all interference/jamming coming from the sectors where the transmitter is not located.

Definition 3: Let R be the communication range of the nodes, JS be the set of jammers, and JR be the jamming range. A link between two nodes A, B is said to be non-jammed if and only:

$$d(A, B) < R \wedge \forall J \in JS : [d(J, B) > JR \vee \text{Sector}(A, B) \neq \text{Sector}(J, B)]$$

Where $\text{Sector}(X, Y)$ denotes the sector used to transmit from X to Y .

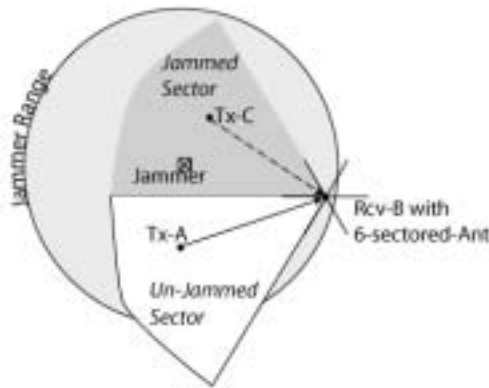


Fig. 3. Node B uses a 6-sectorized antenna. The jammer can only prevent the communication from C to B and not from A to B

As a first step, in the evaluation of the connectivity, we assume that the sectors direction is fixed. As a next step, we will investigate generalization to randomly oriented antennas and steerable antennas.

4.1 Randomly Placed Jammers against Sectorized Antennas

We have simulated the impact of randomly located jammers on the connectivity index when the communicating nodes use sectorized antennas.

To evaluate the connectivity gain achieved by sectorized antennas, we compare the number of jammers that lead to the same connectivity index for 1, 3, 6, and 12 sectors antennas. The simulation area is 2000×2000 and 1000×1000 , the number of nodes is 400 and 200, and communication/jamming range is 200. Note that the 1000×1000 has node density twice that of area 2000×2000 . For a connectivity index of 0.7, a 12 sectorized antenna communication can resist to at least twice the number of jammers leading to a connectivity index of 0.7 for a 6 sectorized antenna. This observation is valid for all values of connectivity index. The simulation seems to indicate that using k -sectorized antennas will lead to resiliency to a factor of k more randomly located jammers. However, were not able to provide a theoretical proof for this simulation result.

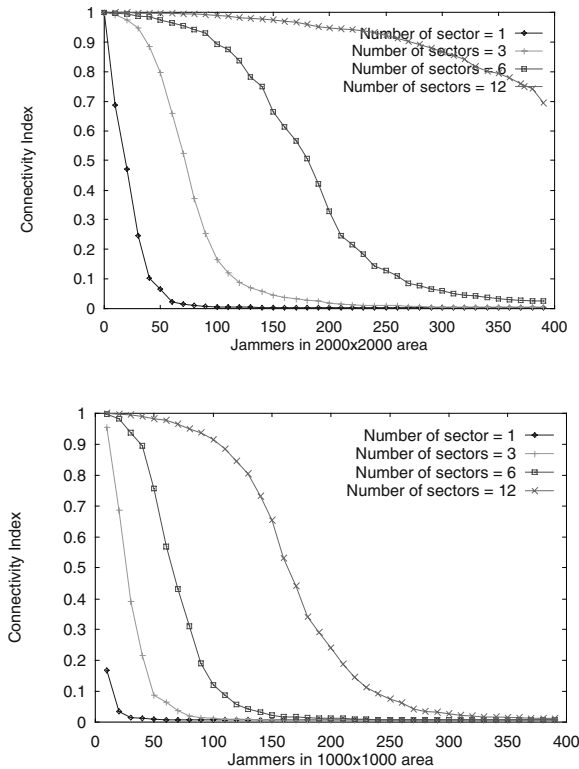


Fig. 4. Connectivity as a function of the number of jammers. The number of communicating nodes is 400 (200) and simulated over an area of $2000 \times 2000 \text{ m}^2$ ($1000 \times 1000 \text{ m}^2$)

4.2 Optimally Placed Jammers against Sectorized Antennas

In this section we analyze the number of jammers required to prevent all communication within an area A when communication nodes use sectorized antennas. We compare this number to the number of jammers required to jam omni-directional antennas.

4.3 3-Sectorized Antennas

We provide an upper bound on the number of jammers needed to prevent, all communication in an area, between nodes equipped with 3-sectorized antenna. The plane is tessellated into equilateral triangles. The jammers are located at the middle of the triangles sides. The sides length is such that for the given jamming range (JR), the jammers centered at points z , t , x , and y satisfy the following property: all circles defined by the jamming limit and centered at z , t , x , and y intersect at the gravity center G of triangle (A, B, C) (See Fig. 5). By considering all area cases it can be shown that any mobile node will be within reach of at least one jammer on each of its sectors.

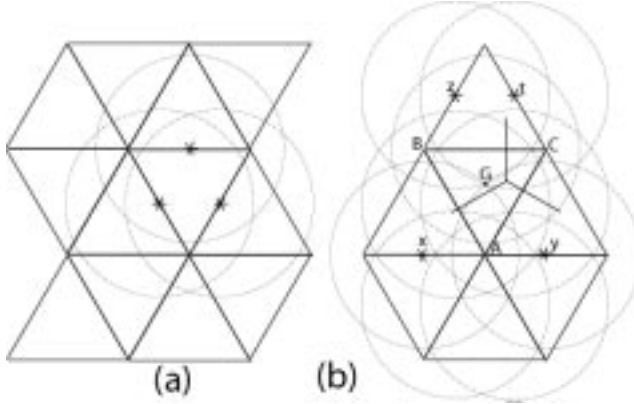


Fig. 5. Jammers placement to jam mobiles equipped with 3-sectorized antennas

Lemma 1: For the circles centered in x , y , z , t (and of radius JR) to intersect at the gravity center of triangle (A, B, C) , the triangle sides has to be equal to $\sqrt{\frac{12}{7}}JR$.

Proof: This can be analytically shown by placing point A at location $(0, 0)$. Therefore the location of points $y = (d/2, 0)$, $G = (0, d/\sqrt{3})$, and $t = (d/4, 3\sqrt{3}d/4)$. Hence

$$\begin{aligned} \text{distance}(y, G) &= JR \\ \sqrt{\frac{7}{12}}d &= JR \end{aligned}$$

It can also be easily shown that $\text{distance}(y, G) = \text{distance}(x, G) = \text{distance}(z, G) = \text{distance}(t, G)$.

Theorem 2: To prevent all communication, between nodes equipped with 3-sectored antennas, within an area A , the adversary need at most 3.5 more jammers in comparison with jamming omni-directional antennas.

Proof: Placing jammers according to Lemma 1 covers all sectors of all nodes. The number of jammers needed is equal to the number of triangles. The number of triangles in a large area A is equal to:

$$\begin{aligned}
 \text{Number of triangles} &= \text{surface}(A)/\text{surface}(\text{triangle}) \\
 &= \frac{4A}{\sqrt{3}d^2} \\
 &= \frac{7A}{3\sqrt{3}JR^2} \quad [\text{From Lemma 1}]
 \end{aligned}$$

From Theorem 1, we know that $\frac{2A}{3\sqrt{3}JR^2}$ is the minimum number of jammers needed

to prevent all communication between nodes using omni-directional antennas within and area A . Therefore, the ratio of number jammers needed against 3-sectored antennas versus omni-directional antennas is at most: $\frac{7A}{3\sqrt{3}JR^2} / \frac{2A}{3\sqrt{3}JR^2} = 3.5$.

4.4 4-Sectored Antennas

Using the same approach as for 3-sectored antennas, and by positioning jammers on the corners of squares of side $JR/\sqrt{2}$. The required number jammers is $A/2JR^2$. Therefore, 4-sectored antennas would require at most $\frac{2A}{JR^2} / \frac{2A}{3\sqrt{3}JR^2} = 3\sqrt{3}$.

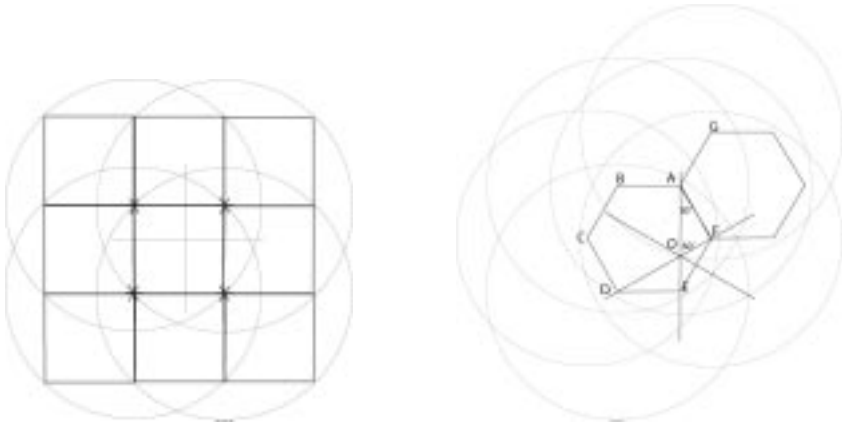


Fig. 6. Jammers location for complete denial of communication against nodes equipped with 4-sectored antennas and 6-sectored antennas

4.5 6-Sectored Antennas

In the case of a 6-sectored antennas, we place the jammers on the vertices of hexagonal cells (A, B, C, D, E, F) as shown in Fig. 6. The jamming radius has to be such that the jammer located on point G can reach point O . The reason for this is to have any point to be covered on all its sectors anywhere within the cell. Using the same approach as 3 and 4-sectored antennas we can conclude that the hexagons sides have to be equal to $JR\sqrt{\frac{3}{13}}$. The number of jammers is $\frac{52A}{9\sqrt{3}JR^2}$. The gain compared to omni-directional antennas is at most: $\frac{52A}{9\sqrt{3}JR^2} / \frac{2A}{3\sqrt{3}JR^2} = \frac{26}{3}$.

	Omni-directional	3-sectors	4-sectors	6-sectors
Required num jammers	$\frac{2A}{3\sqrt{3}JR^2}$	$\frac{7A}{3\sqrt{3}JR^2}$	$\frac{2A}{JR^2}$	$\frac{52A}{9\sqrt{3}JR^2}$
Ratio to omni-directional	1	3.5	$3\sqrt{3}$	$26/3$

Fig. 7. Required number of jammers and the ratio to the required number of jammers against omni-directional communication as a function of the jammed area and jamming range.

We can observe that the gain of multi-sector antennas against optimally placed jammers can be bounded by a value close to the number of sectors. An interesting result would be to derive a general bound on the gain obtained by using N -sector antenna.

5 Mobility Improves Connectivity under Jamming

In this section we investigate how the concept of time-space routing helps against jamming. Here, nodes can buffer a packet until it can forward it to an intermediate node. We are interested in the connectivity of a mobile ad hoc network, under jamming. If the area is not fully jammed, then it is still possible for some nodes to communicate when they are out of the range of the jammers. Two nodes N_1 and N_k are connected if there exists a path $N_1, N_2, N_3, \dots, N_k$ and time instants $t_1 < t_2 < \dots < t_{k-1}$ such that link $N_i - N_{i+1}$ is valid at instant t_i . A full path between two nodes might not exist at a single instant but for each link in the path there should be an instant of time when it is valid. It is natural to expect that mobility will increase the chances of nodes to communicate. If the nodes are moving randomly within an area they will eventually pass through an unjammed area and therefore be able to communicate. In [24] it was already shown that mobility increases capacity of ad hoc networks. We study the connectivity improvement under jamming. Designing a routing protocol that can exploit mobility under jamming using sectored antennas is an important question that we plan to address in the future. As a first step in the analysis, we have simulated a random-walk mobility where the nodes at each step select a random direction and destination within a mobility range. The number of nodes is 400, the jamming nodes vary within

100-400. The simulation area is 2000x2000. At each step the destination is randomly uniformly selected within the disc of radius $R/10$ centered at the current position, which corresponds to a maximum speed of 20m/s. Fig. 8 shows a substantial increase in connectivity when combining sectorized antennas and mobility. Mobility expands the minimum connectivity achieved by sectorized antennas. Although the combination of directivity and mobility increases the connectivity, existing routing and transport protocols are not designed to make use of it. The applications assumptions have also to be reassessed to operate in such environments.

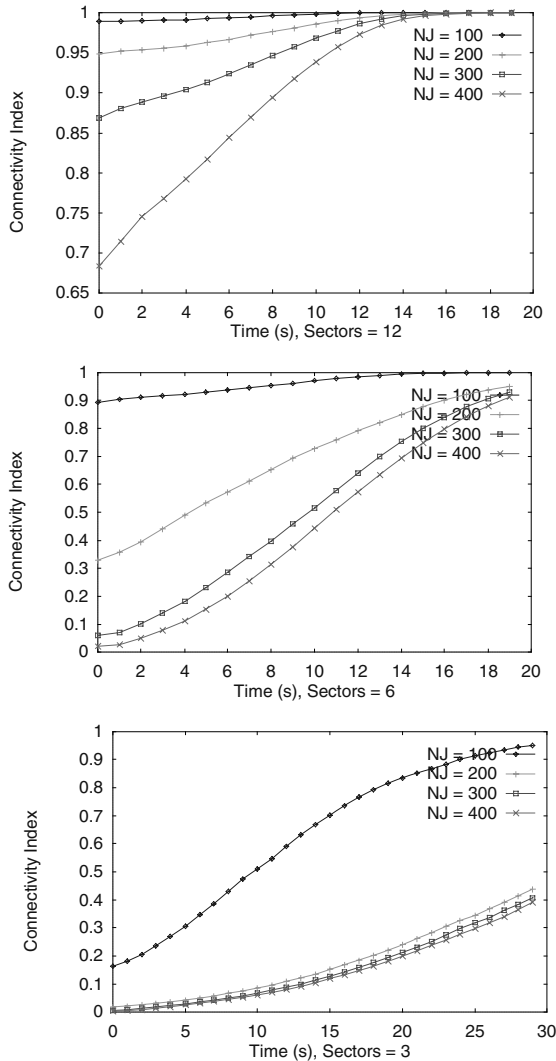


Fig. 8. Connectivity index when combining sectorized antennas and mobility

6 Conclusions and Future Research

In this paper, we have investigated the problem of jamming multihop communication in an ad hoc network. We have shown that even a small number of jammers can drastically reduce the network connectivity when the nodes communicate using omnidirectional antennas. We also showed that a combination of directional antennas and mobility provide significant improvement of connectivity. We investigated both the case where the jammers can be optimally placed by the adversary, and when they are randomly located within an area. Two important problems remain open. First, is there a lower bound on the connectivity gain achieved by a k -sectored antennas versus omnidirectional antennas. Second, existing routing and transport protocols were mainly designed for symmetric or wired networks and would perform poorly in a jammed environment. Therefore, the question on how to design efficient time-space routing protocols and transport protocols that use mobility and take into account the application requirements and temporarily jammed links or areas.

References

1. Guevara Noubir and Guolong Lin. "Low Power DoS Attacks in Data Wireless LANs and Countermeasures". in *Proceedings of Poster: ACM MobiHoc*. 2003. Annapolis, MD: ACM Press.
2. Antenna Ltd. 2003. <http://www.antenova.com/>.
3. Texas Instruments Inc. 2003. www.ti.com/wanda.
4. Hideaki Takagi and L. Kleinrock, "Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals". *IEEE Transactions on Communications*, 1984. **32**(3): p. 246–255.
5. Christian Bettstetter. "On the Minimum Node Degree and Connectivity of a Wireless Multihop Network". in *Proceedings of MobiHoc*. 2002. Lausanne, Switzerland: ACM Press.
6. Miguel Sanchez, Pietro Manzoni, and Z.J. Haas. "Determination of critical transmission range in ad-hoc networks". in *Proceedings of Multiaccess Mobility and Teletraffic for Wireless Communications Workshop*. 1999.
7. Imrich Chlamtac and A. Farago, "A New Approach to the Design and Analysis of Peer-to-Peer Mobile Networks". *ACM/Baltzer Wireless Networks*, 1999. **5**(8).
8. Olivier Dousse, Francois Baccelli, and P. Thiran. "Impact of Interferences on Connectivity in Ad Hoc Networks". in *Proceedings of IEEE Infocom*. 2003.
9. P. Santi and D.M. Blough. "An Evaluation of Connectivity in Mobile Wireless Ad Hoc Networks". in *Proceedings of IEEE DSN*. 2002.
10. P. Gupta and P.R. Kumar, "Critical Power for Asymptotic Connectivity in Wireless Networks". *Stochastic Analysis, Control, Optimization and Applications: A Volume in Honor of W. H. Fleming*, ed. W. M. McEneaney, G. Yin, and Q. Zhang. 1998: Birkhauser.
11. Mathew D. Penrose, "On k -Connectivity for a Geometric Random Graph". *Random Structures and Algorithms*. **15**(2): p. 145–164.
12. Andras Farago. "Graph Theoretic Analysis of Ad Hoc Network Vulnerability". in *Proceedings of WiOpt: Modeling and Optimization in Mobile Ad Hoc and Wireless Networks*. 2003.
13. Yih-Chun Hu, Adrian Perrig, and D.B. Johnson. "Ariadne: A Secure On-Demand Routing Protocol for Ad Hoc Networks". in *Proceedings of ACM Mobicom*. 2002. Atlanta, GA.

14. P. Papadimitratos and Z.J. Haas, *Securing Mobile Ad Hoc Networks*, in *Handbook of Ad Hoc Wireless Networks*, M. Ilyas, Editor. 2002, CRC Press.
15. Pradeep Kyasanur and N. Vaidya, "*Detection and Handling of MAC Layer Misbehavior in Wireless Networks*". August 2002, UIUC.
16. Bridget Dahill, et al., "*A Secure Routing Protocol for Ad Hoc Networks*". 2001, Electrical Engineering and Computer Science, University of Michigan.UM-CS-2001-037,
17. Jean-Pierre Hubaux, Levente Buttyan, and S. Capkun. "*The Quest for Security in Mobile Ad Hoc Networks*." in *Proceedings of MobiHoc'01*. 2001: ACM Press.
18. Sergio Marti, et al. "*Mitigating Routing Misbehavior in Mobile Ad Hoc Networks*". in *Proceedings of Sixth Annual IEEE/ACM International Conference on Mobile Computing and Networking (MobiCom 2000)*. 2000: ACM Press.
19. Frank Stajano and R. Anderson. "*The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks*." in *Proceedings of Security Protocols, 7th International Workshop*. 1999: Lecture Notes in Computer Science, Springer Verlag.
20. Lidong Zhou and Z.J. Haas, "*Securing Ad Hoc Networks*". IEEE Networks Magazine, 1999. **13**(6).
21. Curtis D. Schleher, "*Electronic Warfare in the Information Age*". 1999, Norwood, Artech House.
22. Bernard Sklar, "*Digital Communications, Fundamentals and Applications*". 2001: Prentice-Hall.
23. Richard Kershner, "*The Number of Circles Covering a Set*". American Journal of Mathematics, 1939. **61**(3): p. 665–671.
24. Mathias Grossglauser and D. Tse. "*Mobility Increases the Capacity of Ad Hoc Wireless Networks*". in *Proceedings of IEEE Infocom*. 2001.

The Ad Hoc On-Demand Distance Vector Protocol: An Analytical Model of the Route Acquisition Process

Matthias Hollick, Jens B. Schmitt, Christian Seipl, and Ralf Steinmetz

Multimedia Communications Lab (KOM),
Department of Electrical Engineering and Information Technology,
Darmstadt University of Technology,
Merckstrasse 25, D-64289 Darmstadt, Germany.
{matthias.hollick, jens.schmitt, christian.seipl,
ralf.steinmetz}@kom.tu-darmstadt.de
<http://www.kom.tu-darmstadt.de>

Abstract. Ad hoc networking research suffers from the lack of meaningful and realistic models to describe the route acquisition process of ad hoc routing protocols. There is a strong need for such models to be able to perform realistic calculations supporting important yet difficult tasks, such as performance estimation and protocol scalability analysis. Based on existing work for ideal source routing we formulate and validate an analytical model to match the route acquisition process executed by the Ad Hoc On-Demand Distance Vector (AODV) protocol. This allows us to predict the probability density function of estimated route lengths, a powerful metric for characterization of the network behavior. We further extend our study to include multiple refinements to the basic AODV protocol. The instantiation and validation of the model is completed by means of an experimental analysis.

1 Introduction

The promise of self-organizing operation of mobile and wireless nodes gives rise to several interesting research challenges, of which routing is a very prominent one. A number of experimental protocols have been designed (see, for example, [1] and [2]; Royer presents a fairly comprehensive taxonomy of ad hoc routing in [3]). The main directions of research in this area include performance optimization. Recently, quality of service, security, and scalability issues have also drawn attention. Despite the fact that extensive work is being performed in this area, large-scale ad hoc networks are not currently available for civilian applications.

One key concept of ad hoc routing protocols is the adaptability to constraints induced by mobility and the nature of the wireless medium. Nearly all existing protocols include various performance optimizations, thus complicating the proper analysis of overall network efficiency. There are many simulation-based studies which investigate special but restricted scenarios. To allow for easy generalization of results and further study of scalability related metrics, analytical models describing the detailed behavior of ad hoc routing protocols are of great importance. Only few such models exist, however.

We believe that the availability of precise models is very important for further analysis and optimization of existing and future protocols. Our investigation provides:

1. An analytical model of the route acquisition process used in various ad hoc routing protocols. We include the modeling of transmission errors and exemplify our findings using the AODV [1] protocol.
2. The extension of our model to incorporate various protocol optimizations of AODV. We include *expanding ring search* and *reply by intermediate* [1].
3. The experimental validation of these models.

Our results enable the precise prediction of the route length distribution inside the network which characterizes the overall network behavior. We regard this knowledge as crucial for the study of scalability issues which hinder the further evolution of ad hoc networks to reach a critical mass of wide deployment.

Outline

The next section reviews related work in the area investigated. Sect. 3 introduces the modeling of the ad hoc routing process. We start with an idealized analytical model of ad hoc routing, which we subsequently adapt to fit our prerequisites. Sect. 4 details the modeling of transmission errors. Modeling the AODV protocol is detailed in Sect. 5 while the features *expanding ring search* and *reply by intermediate* are added to the model in Sect. 6 and Sect. 7. The model equations are instantiated and validated by means of simulation. We finish by drawing conclusions and by pointing to possible future work.

2 Related Work

A large number of performance comparisons of various ad hoc routing protocols exist, the work of Das et al. [4] is an early example. These studies are of great importance for verifying exact protocol behavior in well-defined environments. However, they are likely to be imprecise due to the large set of predictor and response variables which need to be considered [5] and cannot deliver qualitative metrics to describe overall network behavior and protocol scalability.

The limitations of simulation studies clearly mandates analytical models which can provide more general results. Early work related to the capacity of multi-hop packet radio networks was performed by Kleinrock et al. (see [6], [7] and [8]). The results account for the link layer performance under various circumstances and are based on a sound analytical approach. They focus on the spatial capacity [6] as well as the optimal transmission ranges for randomly distributed packet radio terminals ([7] and [8]). Recently, the work of Kleinrock et al. was enhanced by Gupta and Kumar in [9], a cornerstone of analytical capacity and performance estimation for large-scale ad hoc networks.

There is, however, little work which takes an analytical approach for describing the realistic characteristics of ad hoc routing protocols. This is especially true if we take protocol optimizations into consideration. The work of Kail, Németh, et al. [10] estimates the possible capacity of ad hoc networks—using a model of the idealized source routing process. In contrast, Santiváñez et al. [11] developed a general performance

comparison metric on an abstract level. Their results account for various principles of protocols. In particular, they studied the complexity of the individual schemes with respect to the induced overhead. Although the work studies the protocol complexity, the characteristic network behavior is not explained further.

Our approach is to obtain deeper insights in the overall network behavior for dedicated protocols—with a higher abstraction level than currently available through simulation. The envisioned model should be realistic enough to describe detailed protocol behavior from the networks perspective. This allows for further application of the model to investigate the influence of problems, such as, node misbehavior on the overall network operation. The route acquisition process constitutes the essential behavior of ad hoc routing protocols. In particular, we have chosen the distribution of route lengths as the metric for our study. Loosely related work in the area of fixed networks which uses the same metric does, however, exist. In [12] Zegura et al. introduce the metrics “length-distribution” and “hop-length-distribution” to compare graph-based models for Internet topology. This metric yields a powerful prediction tool to visualize the overall network characteristics, the interpretation in ad hoc networks being significantly different from within fixed networks. In the following, we describe our model.

3 Modeling of Ideal Source Routing

Basic properties of packet radio networks, which serve as foundation for our work, are provided by Kleinrock and Sylvester [7]. Based on these, our model describes the distribution of route lengths within the network. This metric is used to describe the network behavior. Recently, a model for ideal source routing was studied by Kail, Németh, et al. [10]. The derived equations are not entirely correct, however. In parallel to our work, the excellent work of Miller [13] derived the distribution of link distances in a wireless network. Ref. [13] provides more general results for the link-distance distribution than we do but omits the modeling and validation of realistic protocol behavior. We use the following set of assumptions:

- The investigated area, A , is a normalized square of side length 1.
- The x and y coordinates of the nodes are independently and identically uniformly distributed in the interval $[0,1]$.
- The nodes, N , share a uniform transmission range, r , which is considerably smaller than the side length of the square. The system consists of n nodes.
- The nodes are not in motion.

To describe networks of arbitrary connectivity, property, and size, we make the length parameters dimensionless. Fig. 1 shows the dimensionless network used during modeling. Performing an ex-post analysis of the geometrical node distribution, we find the area, A_0 , one node covers related to the entire area of investigation, A , to be

$$A_0 = \frac{A}{n} . \quad (1)$$

The radius, r_0 , of a circle and side length, b_0 , of a square equivalent to A_0 are

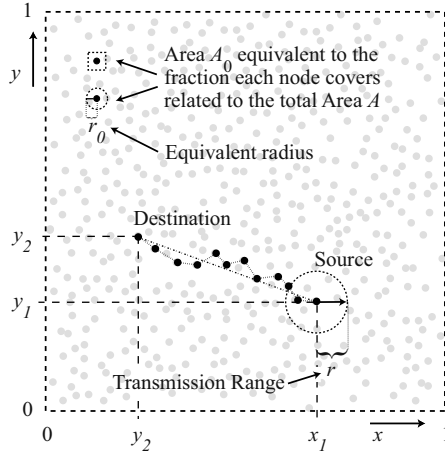


Fig. 1. Normalized sample network. The highlighted nodes, areas, and distances represent the most important properties of the network.

$$r_0 = \sqrt{\frac{A}{n\pi}} \text{ and } b_0 = \sqrt{\frac{A}{n}} \text{ respectively (see Fig. 1).} \quad (2)$$

The average degree, M , of the network (expected number of nodes in a transmission radius at any point) can easily be obtained by

$$M = \frac{\pi r^2}{A_0} = \left(\frac{r}{r_0}\right)^2. \quad (3)$$

It is intuitive that the average number of neighbors equals $M - 1$. Knowing of M allows one to predict how many nodes will be influenced, on average, if one node transmits a signal. We are particularly interested in route lengths. Starting with idealized source routing, a first approximation of the shortest path between two nodes follows the direct line between these, assuming a very large or infinite number of nodes (see Fig. 2 for the corresponding visualization). Thus, the estimated length of hops, h , between two nodes is a function of the geometric distance, d .

The routing protocol uses neighboring nodes to transmit the packets from source to destination. The progress a packet makes in each step can be modeled as follows: Nodes are assumed to be connected directly if they are in range, r , of each other. The median distance, r_1 , between two nodes which can reach each other is

$$r_1 = \frac{r}{\sqrt{2}} \text{ (see Appendix A for the derivation of } r_1 \text{).} \quad (4)$$

If d is sufficiently large compared to r , we can approximate the average progress per routing step as r_1 . As a result, the distance between source and destination is

$$d = h(d)r_1. \quad (5)$$

The geometric distance between two nodes on a plane is also given by the Euclidean distance between their positions. So for Node1 (x_1, y_1) and Node2 (x_2, y_2) the distance is

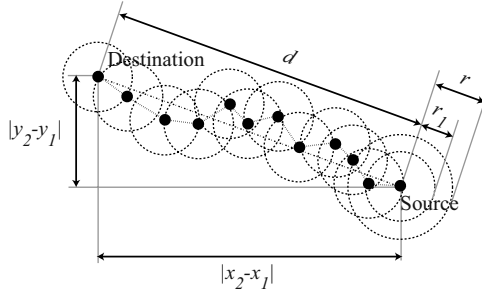


Fig. 2. Geometrical measures within the example network. The distance d between source and destination can be expressed using the hopcount $h=11$ and the radius r_1 .

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (6)$$

We replace the explicit positions in Equation (6) using their distribution and obtain the probability density function $p(d)$ likewise in [13] to be

$$p(d) = \begin{cases} 2d(d^2 - 4d + \pi) & d \leq 1 \\ 8d\sqrt{d^2 - 1} - 2d^3 - 4d + 4d\left(\frac{1}{\sin(1/d)} - \frac{1}{\cos(1/d)}\right) & 1 < d \leq \sqrt{2} \end{cases} \quad (7)$$

This serves as central equation for the remainder of this paper. It describes the statistical relation between the distance of two nodes inside the unit square and the corresponding probability of being connected. The distribution $P(d)$, which will be used to calculate the model predictions, can now be obtained by integration of the probability density function $p(d)$:

$$P(d) = \begin{cases} \frac{1}{2}d^4 - \frac{8}{3}d^3 + \pi & d \leq 1 \\ 4\sqrt{d^2 - 1} + \frac{8}{3}\sqrt{(d^2 - 1)^3} - \frac{1}{2}d^4 + 2d^2 + \frac{1}{3} \\ + 2d^2\left(\frac{1}{\sin(1/d)} - \frac{1}{\cos(1/d)}\right) & 1 < d \leq \sqrt{2} \end{cases} \quad (8)$$

$$pm(d) = \begin{cases} (1 - q)^{h(d)} 2d(d^2 - 4d + \pi) & d \leq 1 \\ (1 - q)^{h(d)} 8d\sqrt{d^2 - 1} - 2d^3 - 4d + 4d\left(\frac{1}{\sin(1/d)} - \frac{1}{\cos(1/d)}\right) & 1 < d \leq \sqrt{2} \end{cases} \quad (9)$$

4 Modeling of Transmission Errors

The basic model assumes that there are no transmission errors on the lower layers (idealized physical and link layer without losses). A wireless medium is, in reality, often shared competitively. To describe this behavior analytically, we relax the error condition by introducing the success probability for each packet transmitted. Our model takes a global approach instead of a local one and sets all success probabilities to be equal (note that under some circumstances the probability may be near zero, as we show later).

For a single hop and a given loss probability, q , the success probability is $1 - q$. If we assume a multihop route consisting of h hops, the success probability is $(1 - q)^h$. Our base model assumes a success probability of 1, thus we need to introduce $(1 - q)^{h(d)}$ respectively $(q)^h$ as a correction term which describes the success probability as a function of $h(d)$, as given in Equation (9).

We are now able to obtain the probability density functions for both; the successfully established routes and, the routes hindered to be established, respectively. We are interested in a direct comparison of the number of routes between the error-free case and the case with errors, in order to predict the routing performance. Thus we give a probability measure, which accounts for the routes remaining unaffected by the error condition (see Equation (9)).

5 Modeling of AODV

We now refine the route length model to include realistic protocols. As a consequence, the routes discovered will differ from the ideal routes. If we take the AODV protocol [1] as an example, we obtain non-optimal routes, due to the loop-freedom criterion.¹ Hence, the forward routing graph which is spanned according to the propagation of the *RREQ* is sub-optimal in the case of AODV. The route length increases due to this effect.

For large networks this elongation of routes may be described by a factor θ . This factor acts multiplicatively on the geometrical distance d . All routes discovered can be described using $d' = \theta d$. Since routes may not be any shorter than the optimal distance, $\theta \geq 1$ for all protocols. θ depends on the routing protocol variant used. The curve resulting from Equation (9) will appear contracted in the y-axis by the factor θ , and stretched in the x-axis by the factor θ . Equipped with this refined model, we are now able to directly compare the analytical results with simulation results, using θ as correction term for the protocol chosen. We thus obtain $h(d) = d'/r_1 = d\theta/r_1 = d(\theta/r_1) = d/r'_1$ where $r'_1 = r_1/\theta$.

5.1 Experimental Validation of the Basic Model

To allow for experimental validation, we extended the Qualnet[®] network simulator to comply with a recent AODV draft. Tab. 2 provides an overview of the parameter set for

1. A node which received a route request (RREQ) on a longer but faster way forwards this non-optimal request.

all simulations used for experimental validation within this paper. We use IEEE 802.11b in ad hoc mode as lower layer protocol [14].

The model assumptions for the basic model have been validated using Test1. Since we are mainly interested in investigating the distribution of route lengths, we generated a series of single packets. The rationale behind this configuration is to trigger route discoveries without loading the network unnecessarily. Using the AODV protocol as a predictor variable, the simulations validate the hypothesis of the route length distribution. Moreover, we obtain a first estimate for θ . The possible *reply by intermediate* (see next Section) was avoided by setting the pause time between the individual route requests to 10 seconds. AODV invalidates the cached reverse paths within this period. We measured the length and number of valid routes as response variable. The results are given in Fig. 3. The mean values were obtained in 20 simulation runs with different seeds. Unless stated otherwise, we also calculated and present the estimated standard deviation and the two-sided 95% confidence interval of all the data obtained experimentally.

We use the least-squares method to obtain the fitting parameter, θ , of the curve described by Equation (9) for the measured data. Routes longer than $h = 22$ hops for AODV are not used in the fitting. These routes can be considered unstable and only account for less than 5% of all routes acquired. The application of Equation (9) produces Fig. 3. We obtain a good fit above $h = 5$ hops, while for small h , the measured values are too high. This can be explained by the average transmission range. Our initial assumption was that the number of hops, h , multiplied by the average distance between nodes, r_1 , equals the estimated distance. For routes longer than 5 hops this holds. For neighboring nodes, however, the destination will answer directly even if outside the circle of radius r_1 . This special behavior can be observed for routes up to approximately 5 hops. In theory one would need to model the average range r_1 as function of h . For the case $h > 5$, we expect $r_1(h) \approx r_1$; for the case $h = 1$, $r_1(h) = \sqrt{2}r_1 = r$. For the sake of simplicity, we omit the modeling of this special behavior in the remainder of this paper. The fitting parameter, θ , as well as q , both obtained using the simulation, are as follows: Test1: $\theta = 1.2$, $1 - q = 0.99$.

6 Modeling “Expanding Ring Search”

Expanding ring search is a protocol optimization which AODV uses to increase the protocol efficiency. Given the assumption that the communicating nodes are located nearby, the pure flooding of route requests would generate an unnecessary amount of network traffic. *Expanding ring search* is a stepwise increase of the time-to-live of routing requests (*RREQ*). The *RREQ* is first propagated with hop count 1. If no route is found, the hop count is increased to 3, then 5 and then 7. As the propagation boundary increases, the network load increases as well. The limited propagation over the short distance produces nearly optimal graphs. As soon as the *RREQ* is flooded throughout the network, contention for the medium may introduce additional errors. This hinders an optimal propagation and, as a result, the graph (spanning tree) degenerates. This needs to be considered within the model in two ways. Firstly, the propagation needs to be divided into two distinct areas. For the area covered by the *expanding ring search*, h_{ers} , we obtain θ_{ers} ; for the wider area we obtain θ_f as the correction factors. Secondly, the net-

$$pm(d) = \begin{cases} (1 - q_{ers})^{h(d)} 2d(d^2 - 4d + \pi) & d \leq r_1 \frac{h_{ers}}{\Theta_{ers}} \\ (1 - q_f)^{h(d)} 2d(d^2 - 4d + \pi) & (h_{ers} + 1) \frac{r_1}{\Theta_f} \leq d \leq 1 \\ (1 - q_f)^{h(d)} 8d\sqrt{d^2 - 1} - 2d^3 - 4d \\ + 4d \left(\frac{1}{\sin(1/d)} - \frac{1}{\cos(1/d)} \right) & 1 < d \leq \sqrt{2} \end{cases} \quad (10)$$

$$\Theta(d) = \begin{cases} \Theta_{ers} & d \leq r_1 \frac{h_{ers}}{\Theta_{ers}} \\ \Theta_f & (h_{ers} + 1) \frac{r_1}{\Theta_f} \leq d \leq \sqrt{2} \end{cases} \quad (11)$$

$$d' = d\Theta(d) \quad (12)$$

$$h = \frac{d\Theta(d)}{r_1} \quad (13)$$

work load is influenced. If the *expanding ring search* is successful, the overall network load is reduced and thus the error probability decreases. For hop counts larger than h_{ers} , the errors follow the model introduced in Sect. 4.

As a result, we obtain a function with 3 sections. Consequently we only investigate the case where $h_{ers}r_1 < 1$, since the transition to flooding will usually be smaller than the normalized distance $d = 1$. The corresponding model equations are Equation (10)-(13). Please note that the equations are only valid if appropriate fitting is performed.

6.1 Experimental Validation of “Expanding Ring Search”

For experimental validation, we conducted the experiment Test2 to examine the behavior of AODV with *expanding ring search* activated (see Tab. 2 for parameters). The remainder of the simulation parameters is set identical to the ones used to validate the base model (Test1). We carried out 20 replications for the experiment.

The results for Test2 are shown in Fig. 4. We see a significant increase in routes in the close vicinity. Moreover, the steps of *expanding ring search* are clearly visible for AODV. The parameters for *expanding ring search* are set to $TTL_START=1$ and $TTL_INCREMENT=2$. The upper bound for the search is $TTL_THRESHOLD=7$. The *RREQ* will thus be flooded if no route is found using TTLs of 1, 3, 5 and 7. For our model, these increments induce different network loads and thus a stepwise function for the area $h = 1$, $1 < h \leq 3$, $3 < h \leq 5$, $5 < h \leq 7$, and $h > 7$. The end of the curve is a result of *RREQ* flooding. For the sake of simplicity, we considered only the two main segments of the curve within Equations (9) and (11). The parameters obtained for these two segments are as follows (see also Tab. 1): Test2: $\Theta_{ers} = 1.04$, $\Theta_f = 1.19$, $1 - q_{ers} = 0.958$, $1 - q_f = 0.98$.

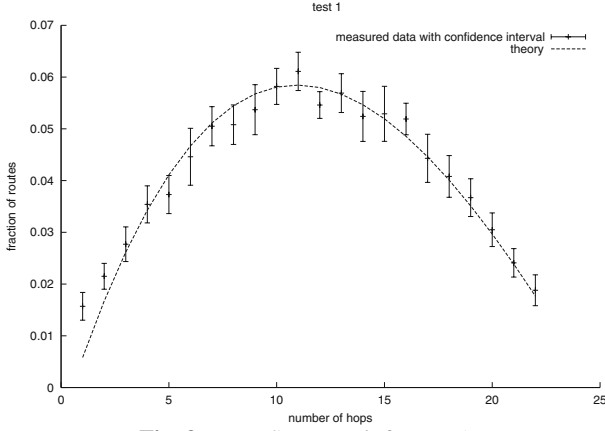


Fig. 3. Least-Squares Fit for Test1.

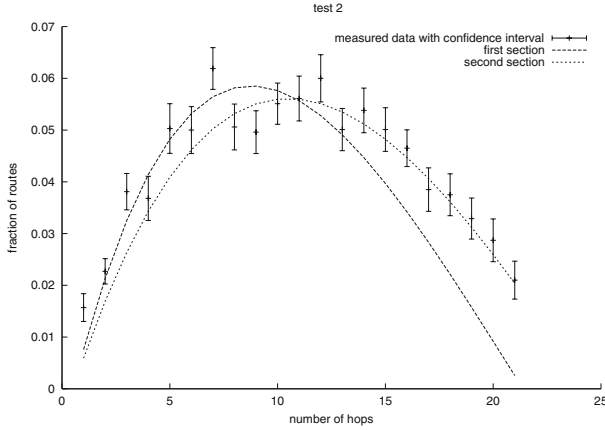


Fig. 4. Least-Squares Fit for Test2.

7 Modeling “Reply by Intermediate”

Another feature of AODV is the possibility that intermediate nodes with valid routes answer the *RREQ*. This feature firstly may shorten the duration of the routing cycle. Secondly, the probability of acquiring longer routes is heightened and thirdly, the resulting routes may be prolonged under special circumstances.

The correction term introduced in Sect. 4 needs to be adopted according to *reply by intermediate*. If we assume the length is reduced by half, we obtain $(1 - q)^{h(d)/2}$ instead of $(1 - q)^{h(d)}$. In general, route reductions from $h(d)$ to $\sigma h(d)$ (with $\sigma < 1$ as the correction term) can be modeled by $(1 - q)^{\sigma h(d)} = ((1 - q)^{\sigma})^{h(d)}$. The measurement of σ on the right side of the equation is not trivial, because we are only able to calculate the combination of σ and q from our experiments. The value of $(1 - q)^{\sigma}$ may be obtained by knowing h . For an exact estimation of σ , we need to know the distance at which the replying node resides. Nevertheless, the results are suitable for our purpose. Quantitatively, we see a decreasing error probability and a subsequent increase in longer routes.

Table 1. Results of Routing Related Simulations

Test	AODV		
	Test1	Test2	Test3
θ_{ers}	n.a.	1.04	1.064
θ_f	1.2	1.19	1.19
$1 - q_{ers}$	n.a.	0.958	0.976
$1 - q_f$	0.99	0.98	0.98

7.1 Experimental Validation of “Reply by Intermediate”

The experimental validation is performed with Test3 to examine the behavior of AODV. To allow for a greater number of active routes, we increased the number of *RREQs* while the rest of the simulation parameters was kept the same as previously used. The experiments were performed using 20 replications each.

The results for Test3 are depicted in Fig. 5. We notice a better reply behavior due to the increased activity and possible replies by intermediate nodes. Moreover, the *expanding ring search* characteristics produce two sections of the curve which are clearly visible. The combination with *reply by intermediate* gives a smoother transition between the individual steps, thus supporting our approach of modeling the whole equation in two steps. The fitting of the curves is also given in Fig. 5 for Test3. The measured values and fitting parameters for the experiments are listed in Tab. 1.

7.2 Summary of Results

We present the summary of results of our simulations in Tab. 1. The success probability of finding a valid route is larger using *expanding ring search* and *reply by intermediate* than using pure AODV. The comparison of Test2 and Test3 reveals a remarkable result. One would generally expect that the much higher network load (by factor 20) should result in lower success probabilities. Since the increased traffic on the other hand allows

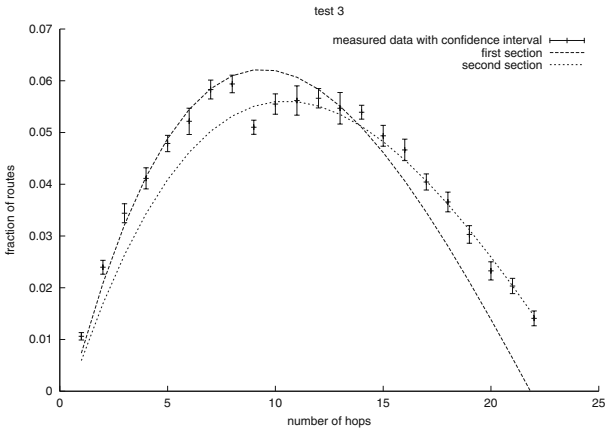


Fig. 5. Least-Squares Fit for Test3.

Table 2. Experimental Parameter Set

Test	Test1 (AODV)	Test2 (AODV)	Test3 (AODV)
Simulation area	$(3300.94\text{m})^2$	$(3300.94\text{m})^2$	$(3300.94\text{m})^2$
Number of nodes	500s	500	500
Duration	5050s	5050s	1500s
Replications	20	20	20
Mobility	no	no	no
Expanding ring search	no	yes	yes
Traffic	every 10s one stream	every 10s one stream	every 500ms one stream
Packets (per flow)	1	1	1
r_0	83.287m	83.287m	83.287m
r_l	176.679m	176.679m	176.679m
M	9	9	9
Other parameters	Transmission Power = 7dBm; Propagation Model = Free Space; Transmission Range (r) = 249.862m; MAC 802.11b DCF; Max. Transmission Rate = 11 MBits/s; Local Repair = Deactivated; Hello Messages = Deactivated; Packet Size = 512Byte; UDP as Transport protocol		

for replies by intermediate nodes, the probability is nearly similar and in Test3 even above the result with low network load.

8 Conclusions

We have discussed the realistic modeling of ad hoc routing protocols. As a first step, we analytically modeled the route acquisition process. Hereby, our model predicts the route length distribution within the network. We extended the idealistic assumptions of existing models to cover transmission errors. We then modeled the behavior of the AODV protocol. To reflect more realistic protocol behavior, the AODV features *expanding ring search* and *reply by intermediate* were integrated as well.

We validated all models presented within this investigation by means of experimental analysis. Our findings are that our model gives precise predictions of the route length distribution within ad hoc networks operating with realistic protocols. Our model aids to realistically analyze routing performance within the area of ad hoc networking, thus allowing to serve for various purposes in scalability / performance estimation. As future work, we perceive the improvement of currently available routing protocols by, for example, quantifying the effect of node misbehavior with support of our model [15].

References

- [1] C. E. Perkins, E. M. Belding-Royer, and S. R. Das. Ad hoc On-Demand Distance Vector (AODV) Routing. Experimental, RFC 3561, July 2003.
- [2] R. Ogier, F. Templin, and M. Lewis. Topology Dissemination Based on Reverse-Path Forwarding (TBRPF). Internet Draft, draft-ietf-manet-tbrpf-10.txt, July 2003.
- [3] E. M. Royer and C.-K. Toh. A Review of Current Routing Protocols for Ad Hoc Mobile Wireless Networks. *IEEE Personal Communications*, 6(2):46–55, April 1999.

- [4] S. R. Das, C. E. Perkins, and E. M. Royer. Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks. In *Proceedings of Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM'00, Tel Aviv, Israel*, volume 1, pages 3–12, March 2000.
- [5] R. Jain. *The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling*. Wiley-Interscience, New York, NY, USA, May 1991.
- [6] R. Nelson and L. Kleinrock. The Spatial Capacity of a Slotted ALOHA Multihop Packet Radio Network with Capture. *IEEE Transactions on Communications*, 32(6):684–694, June 1984.
- [7] L. Kleinrock and J. Silvester. Optimum Transmission Radii in Packet Radio Networks or Why Six is a Magic Number. In *Proceedings of National Telecommunications Conference, Birmingham, AL, USA*, pages 4.3.1–1.3.5, December 1978.
- [8] H. Takagi and L. Kleinrock. Optimal Transmission Ranges for Randomly Distributed Packet Radio Terminals. *IEEE Transactions on Communications*, 32(3):246–257, March 1984.
- [9] P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, March 2000.
- [10] E. Kail, G. Németh, and Z. R. Turányi. The Effect of the Transmission Range on the Capacity of Ideal Ad Hoc Networks. In *Proceedings of the 4th International Symposium on Wireless Personal Multimedia Communications (WPMC'01)*, September 2001.
- [11] C. A. Santiv  ez, B. McDonald, I. Stavrakakis, and R. Ramanathan. On the Scalability of Ad Hoc Routing Protocols. In *Proceedings of Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies, INFOCOM'02, New York, NY, USA*, volume 3, pages 1688–1697, June 2002.
- [12] E. W. Zegura, K. L. Calvert, and M. J. Donahoo. A Quantitative Comparison of Graph-based Models for Internet Topology. *IEEE/ACM Transactions on Networking*, 5(6):770–783, December 1997.
- [13] L. E. Miller. Distribution of Link Distances in a Wireless Network. *Journal of Research of the National Institute of Standards and Technology*, 106(2):401–412, March–April 2001.
- [14] IEEE Computer Society LAN MAN Standards Committee. Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1997. IEEE Std. 802.11-1997.
- [15] M. Hollick and J. Schmitt. On the Effect of Node Misbehavior in Ad Hoc Networks. Technical Report TR-KOM-2003-07, Darmstadt University of Technology, Multimedia Communications Lab, 2003. Available at <ftp://ftp.kom.tu-darmstadt.de/pub/TR/TR-KOM-2003-07.pdf>.

Appendix A—Derivation of r_1

Let A_n denote the area a node covers with its radio:

$$A_n = \pi r^2 \quad (14)$$

Given a random distribution of all nodes located within this area, we calculate the area which hosts 50% of the nodes:

$$\frac{A_n}{2} = \frac{\pi r^2}{2} \quad (15)$$

The radius covering this area is:

$$r_1 = \sqrt{\frac{r^2}{2}} = \frac{r}{\sqrt{2}} \quad (16)$$

This radius r_1 describes the median distance between two neighboring nodes.

A Scheduling Algorithm for a QoS Based Satellite Network

Abheek Saha

Hughes Software Systems, Plot 31, Sector 18, HUDA Electronic City, Gurgaon, Haryana,
INDIA
asaha@hss.hns.com

This paper describes a new scheduling algorithm for multimedia applications on wireless networks. The scheduling algorithm is based on standard GPS class scheduling algorithms; however instead of operating on a single QoS attribute, it can use as input two variables, one governing the minimum committed rate and the other governing the access to surplus bandwidth. The scheduling algorithm also does not assume full control over the transmission because of optimizations made by the transmission modules to maximize achieved capacity.

1 Introduction

Resource Allocation for Data Networks

The resource allocation function in a wireless system divides the available radio-resources to the users who need these resources. Depending on the nature of the air-interface, the resource allocation can be at different levels of dynamicity. Modern systems, such as GPRS, UMTS and 3rd generation satellite based systems use fully dynamic resource allocation mechanisms (also known as bandwidth-on-demand). This has multiple advantages. It allows the user to be ‘always on’, without requiring system resources to be expended on her; it also allows the operator to put into place usage based billing. Further, it significantly increases the capacity of the system by allowing a significant number of users running a mix of different applications to share a single frequency channel at a very fine granularity. Allocation of resources are made as and when there is demand for the same; this allows an optimal operating point of performance and capacity to be maintained.

The key to this function is the resource allocation or scheduling algorithm, which controls access to resources. The resource allocation algorithm has to support different applications with their individual load generation patterns. Applications that are delay-sensitive (such as voice and interactive traffic) has different needs than applications that are throughput bound. Different applications have different rate control mechanisms, which use different network parameters to ‘sense’ the network conditions. Finally, with the advent of QoS and subscription services, the scheduling algorithm has to be able to allocate resources based on pre-negotiated QoS guarantees, yet maximize the perceived aggregate performance and system capacity.

Standard Scheduling Algorithms

Standard scheduling algorithms have been studied for a fair amount of time; however, interest in these algorithms have been rising as the Internet enters the next phase of its

existence. The need to support multiple services (voice, video and data transfer) as well as the need to offer differentiation in services to different customers have led to the renewed interest in various kinds of scheduling algorithms. The key property of a scheduling algorithm is its 'fairness'; i.e. the way it distributes resources to different flows.

Many currently available algorithms are based on the Generalized Processing Sharing (GPS) class of algorithms. . The GPS algorithms have many advantageous properties; they converge to proportional fairness and, for controlled flows, they can guarantee bounds on per packet delay independent on the mix of other traffic flows in the system. Different algorithms of the GPS class are available, such as Weighted Fair Queueing, Self Clocked Fair Queueing [4] and Worst case Fair, Weighted Fair Queueing [5] .

Many typical scheduling algorithms work as follows. Traffic is classified into flows (one or more than one flows per user). Each flow has an associated QoS attribute. The attribute may be a relative number, or an absolute number specifying the amount of bandwidth to be given to this flow. Based on this attribute, each incoming packet is assigned an order of service. Packets are transmitted based on this order.

One drawback of the above is that the scheduling algorithm works with a single QoS attribute. The attribute can be an absolute number (i.e.a minimum committed rate) or a relative number (a guaranteed share of the bandwidth). If an absolute number, surplus bandwidth in the system is shared out in proportion to the minimum committed rate. This is clearly undesirable. Consider, for example, a voice connection and a packet connection sharing a link. The voice connection requires a guaranteed rate, but has no need of surplus bandwidth. The packet rate can live with a very low or no guaranteed rate, but would like to access all the surplus bandwidth.

[6] and related works have proposed an algorithm called Hierarchical Fair Service Curve, which attempts to solve this problem, by utilizing the concept of a service curve. A service curve need not necessarily be linear; thus for low demand, it will focus on minimizing delay and for high bandwidth work, it will work by minimizing throughput.

A second scheduling algorithm which explicitly handles dual QoS attributes is Fair Share Queueing, proposed by Shenker Scott [7] . Fair Share Queueing optimizes the delay-bandwidth product; thus higher bandwidth applications naturally see lower delay and vice-versa. Fair Share queueing has very strong fairness properties and very strong stability properties as well for adaptive applications; however it moves control of QoS away from the user to the network design.

The scheduling algorithm we have designed is simplified from the ideas of a service curve. However, instead of working with a delay-bandwidth duo, we use the combination of a guaranteed rate and traffic handling priority. The delay guarantees are achieved by appropriate provisioning of the QoS i.e. by selecting the appropriate QoS settings.

Wireless Scheduling Algorithms

Wireless scheduling algorithms are further adaptations of standard scheduling mechanisms as discussed above. The distinction between the two is that wireless schedulers deal with packet drops as a part of the environment. Packet dropping

causes reduced utilization of the radio-resources for a connection; it also causes protocol level effects such as stalling. In summary, two connections with the same QoS subscription, being given the same resources will see different levels of performance, due to location dependent errors and user mobility.

The wireless scheduling algorithms thus are designed to deal with ‘external unfairness’. Regardless of whether the unfairness is due to location dependent errors or due to modifications to the resource allocation pattern due to external modules, the scheduling algorithm tracks and tries to adjust for these issues.

This concept is applicable to our case, due to a separation between the scheduler and the transmission mechanism, as shown below. Using the lead-lag concept, one can correct for unfairnesses caused by the burst-building and transmission algorithms and correct them. Further work remains to be done on this subject.

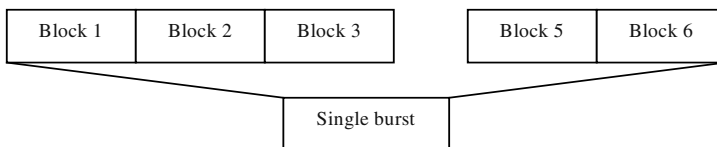
Overview of the System

The design of the packet scheduler has been driven by the features of the target system. The the target system is a UMTS derived satellite system, to be launched world-wide, offering personal voice and data communication services to portable and mobile terminals over most of the world. While the link and higher layers are based on standard UMTS, the physical layer is TDMA based and significantly different from WCDMA systems in its characteristics. In the rest of this section, we define these features and their impact.

The Physical Layer

The system physical layer uses an FDM mode of transmission to multiple users. Each burst covers a ‘frame’ of a fixed duration and contains multiple blocks of a fixed duration and physical symbols. Each block has its own FEC coding; the FEC coding is chosen based on the link conditions of the target UTs; i.e. the UTs for whom data is encoded in the block, and thus have different user-payloads. The capacity of the burst is thus the sum of the realized payloads of the blocks and depends on the relative mixtures of user terminals.

This means that depending on the current link conditions, different sources utilize the same amount of physical resource at different levels of efficiency; obviously, a source with a higher FEC coding rate will be able to transmit more user payload in the same resource as a source with a more robust coding rate. We call this the Physical Capability Index or PCI of the source.



The Quality of Service Paradigm

The quality of service paradigm was chosen to be application friendly, simple and easy to use. It uses a combination of three parameters

- The minimum guaranteed bandwidth (MGB) is an absolute bandwidth (in kilobits/second) which guarantees a given user a given amount of bandwidth if she has sufficient demand. This is a user-specific number and is part of the user's subscription information.
- The traffic handling priority (THP) is an index which describes the users relative priority for 'surplus' resources. Surplus resources are defined to be those which are left over after meeting the MGB for all backlogged users. This is also a user-specific number and is part of the user's subscription information.
- Traffic delay sensitivity (TDS) is a flag which defines whether the application is delay-sensitive i.e. circuit/packet switched voice, ISDN, video, etc. or not. This information is specific to an application type and not to a user. *Currently the TDS flag is not used in the scheduling algorithm.* However, by appropriate setting of the QoS paradigms, it is possible to honour delay sensitivities of individual applications; this can be seen by looking at the sample results.

One of the fundamental decisions that drove the design of the scheduling algorithm was on the subject of 'penalizing' under-performing users i.e. users with a lesser PCI value. One of the objectives of the scheduler is to maximize the overall realized capacity of a channel; i.e. use it at maximal level of efficiency. Since this is best achieved by users operating at the best possible link configuration, it was felt that there should be an incentive for users to point their antennae correctly and otherwise configure the system to get the best performance. Thus, rather than compensate underperformers, it was necessary to give them a reason to retune. On the other hand, a large fraction of the terminals sold were smaller terminals, which by their very design and the nature of the antennae used could not use very aggressive coding schemes for transmission. It was important that these terminals be usable i.e. not be entirely swamped by advantaged terminals in a system. Thus the Quality of Service paradigm has to provide minimum performance to all terminals (regardless of link condition) as well as be able to give strong differentiation for terminals with different link conditions.

Burst Building

The combination of user data PDUs into blocks and then into a single burst is carried out by a burst building module, which is separate from the scheduler. This burst building module uses a layout optimization algorithm which fits in the submitted data to it in an optimal way to maximize the amount of user data carried in each burst. This capacity optimization step needs to have some flexibility in terms of the mix of users to be serviced and the number of bytes to be serviced in order to fill individual blocks in the optimal manner. The exact transmission plan is determined on the fly by the layout optimization algorithm and is out of control of the scheduling engine.

2 Design of the Scheduling Algorithm

Challenges in the Design

Based on the discussion above, we can see that our scheduler implementation has to face the following challenges:

- a) It has to handle two QoS attributes for each terminal and allocate resources based on both the attributes.
- b) It does not have direct control over the transmission sequence, unlike normal scheduling implementations. The final sequence is determined by the layout algorithm, which may move the PDUs around to optimize the transmission capacity.
- c) The scheduler runs periodically (once every frame) and has to provide enough information for the burst-building module to build the frame with maximum usage of the system.
- d) The exact capacity of a given burst is not known, because the coding rates to be used for individual blocks are not known apriori. Thus the initial capacity and bandwidth availability is a guess.
- e) Individual terminals have different physical capabilities and the resource allocation has to simultaneously guarantee them a minimum rate as well as provide them an incentive to retune their systems.

The Actual Design

Based on our discussions above, we have based the design of the scheduler on a standard scheduling algorithm [WF2Q]. We have used the self-clocking design suggested in [4].

Interface to the Layout Algorithm

The scheduling algorithm only runs at fixed intervals and the burst-building algorithm has to be able to modify the exact transmission order (in order to build the burst efficiently). Thus, the burst-building algorithm needs a sufficient 'look-ahead' in order to have the required flexibility in combining PDUs to make a burst. The design of the scheduling algorithm thus is in two parts. In part one, it creates a target list of PDUs for service and hands it over to the burst-building function. This list is ordered according to the ideal service order and based on the information currently known to the scheduler. Once the burst is built, the scheduling algorithm looks at the outcome and adjusts its internal estimates of start and end-times accordingly.

In our implementation, the layout optimizer is given two lists of PDUs for each flow. The first list (MINLIST) is the list of PDUs which must be serviced. The second list, the MAXLIST (a superset of the MINLIST) is the maximum number of PDUs which may be serviced from this flow. The layout algorithm will first fill in the MINLIST for all flows and then the MAXLIST for all flows. It is guaranteed to service PDUs in the order in which they are submitted (within a flow). It will also let the scheduling algorithm know which PDUs have been serviced.

The Scheduler

The scheduling algorithm works by maintaining two sets of virtual start times and finish times. Other than the standard starting and finish times, we also maintain an Estimate Finish Time (EFT) and an Estimated Start time (EST). Whereas the start and finish times do not take into account the surplus bandwidth available, the EFT and EST do take this into account. By looking at the current list of backlogged connections, the scheduler can compute the surplus bandwidth available in the channel and thus compute an Estimated Current Bandwidth (ECB) for each source.

The MINLIST is composed of PDUs whose VFTmin will expire within the next scheduling period. MAXLIST is composed of PDUs whose EFT will expire within the next scheduling period.

Scheduler Operations

There are three main scheduler operations:

- a) When a new packet is received, it is entered into the queue the associated target times are computed. This is handled by the *Enqueue()* function.
- b) The list of PDUs to be serviced are scheduled regularly, using the *createMinList()* and *createMaxList()* functions.
- c) A final cleanup takes place when the actual transmission is completed.

The key variables are as follows:

$Q_m.q$ = queue associated with flow q_m .

$Q_m.q.le$ = last packet in $Q_m.q$

$Q_m.vst$ = virtual starting time associated with first packet in the queue for flow q_m

$Q_m.vftmin$ = virtual finishing time based on the MGB

$Q_m.vftmax$ = virtual finishing time, based on both the MGB and the THP

$Q_m.est[j]$ = estimated start time for the j th packet

$f.pktsz$ = packet size of the packet f

$f.vst$ = ideal starting time for packet f

$f.vftMin$ = ideal finishing time for f based on the MGB of the associated flow

$f.vftMax$ = ideal finishing time for packet f based on the MGP and the THP of the associated flow

V = current virtual time

$B(t)$ is the set of backlogged flows at time t

$MGB[q_m]$ = minimum guaranteed bandwidth associated with flow q_m

$THP[q_m]$ = traffic handling priority associated with flow q_m

$PCI[q_m]$ = Physical Capability Index associated with flow q_m

$ECB[q_m]$ = estimated current bandwidth available to flow q_m , based on its MGP, its THP and the current set of backlogged flows $B(t)$

The key functions are described below.

```

Enqueue(pkt_t *f)
{
    
$$\text{aggminbw} = \sum_{i \in B()} \text{MGB}[i] / \text{PCI}[i]$$

    qm = flow associated with f ;
    if (qm.q is empty)
    {
        qm.vst = V;
        qm.vftMin=qm.vst+f.pktsize/MGB[qm] ;
        update V ;
        
$$\text{aggpriority} = \sum_{i \in B()} \text{THP}[i] ;$$

        surplus = EstCap - aggminBw ;
        ECB[qm] = surplus*THP[qm]/aggpriority;
        qm.EST[f] = qm.vftMAX[qm.q.lE] ;
        qm.vftMax=qm.EST[f] +
            f.pktsize/(MGB[qm]+ECB[qm]) ;
        add packet f to qm.q
    }
}

```

Note that we have computed a single start time for each packet and two finishing times.

There is no direct dequeue function as in a conventional queueing algorithm. The potential PDUs for transmission are created in the *createList()* functions. These function are shown below. Note that these PDUs are not physically dequeued, but pointers to these lists are passed to the layout function, which moves the contents to the output frame for transmission.

```

createMinList()
{
    for( not finished)
    {
        qm = flow with minimum VFTmin  $\forall$  q.VST < V ;
        if (qm == NULL) break ;
        addToList(f=first packet from qm.q) ;
        if (qm.q is not empty)
        {
            f.vst = V;
            f.vftMin=f.vst+f.pktsize/MGB[qm] ;
            update V ;
        }
    }
}

```

```

createMaxList()
{
    for (!finished)
    {
        qm= flow with qm.est < V &&
            min (qm.VFTmax over all backlogged
                queues) ;
        addToList(f=next packet from qm.q) ;
        if (qm == NULL) break ;
        if (this is not the last packet in the flow)
        {
            qm.EST = qm.VFTmax ;
            aggminbw =  $\sum_{i \in B()} MGB[i] / PCI[i]$ 
            aggpriority =  $\sum_{i \in B()} THP[i]$  ;
            surplus = EstCap - aggminBw ;
            ECB[qm] = surplus*THP[qm]/aggpriority
                    + MGB[qm] ;
            f.EST = qm.vst ;
            f.vftMax = f.EST + f.pktsize/(ECB[f]) ;
        }
    }
}

```

The actual dequeue takes place once the transmission has completed and the layout function returns the list of PDUs actually transmitted to the scheduling function. This is handled in the ‘cleanup’ stage.

The algorithm can become very complex due to repeated computation of the surplus throughput. In a standard WFQ algorithm implementation, the virtual times have to be recomputed for a given queue when there is a change in that queue status. However, in this algorithm, the surplus is affected whenever any queue changes from backlogged state to idle or vice versa. As a compromise, we compute the surplus only once every frame boundary. This reduces the computational complexity without any perceptible affect on the output.

3 Testing the Scheduling Algorithm

We have tested our scheduling algorithm using a simulated test bed built around the Opnet tool. In the absence of a single definitive test-suite, the Opnet tool gave us access to a rich set of applications and transport layer implementations, and the ability to implement and test the design at a very fine level of accuracy with relatively less implementation effort.

The test-bed can make measurements at two levels. For one, it can measure performance for each transaction; this obviously varies based on the application type.

Thus, for FTP transactions, we measure the average throughput (the transaction size divided by the transaction time), for HTTP transactions we measure the average page response time, and for CS connections we measure delay and delay-variance.

Secondly, at the level of individual connections, we collect and compute some general statistics. The three key parameters are

the arrival and service rates processes. We model the backlog of each transaction, as well as the burstiness of the input and output for each connection using the index of dispersion

The fairness of scheduling at a frame by frame level.

Choice of Environment and Traffic Models

Based on the classification in [8] and our system requirements, we modelled four kinds of traffic through the scheduler.

- a) CBR traffic. These sources generate a fixed amount of data periodically, with very small jitter. This model represents constant bit-rate voice and ISDN traffic.
- b) Rate variable VBR traffic. These sources generate data at a fixed interval, but the amount of data generated is very variable. This kind of traffic would include video streams.
- c) Rate adaptive traffic. This is a representation of the TCP transport protocol. The model represents an AIMD rate adaptive source, where the trigger for rate decrease is when the queue at the scheduler crosses a particular threshold i.e. tail-drop queueing.
- d) Delay adaptive traffic. This models traffic that is sensitive to end-to-end delay. Examples of delay sensitive transport protocols include modern TCP stacks like TCP Vegas or TCP Westwood. The transfer rate decays exponentially against the fraction of queueing delay in the overall measured round-trip delay seen by the source.

Results

A set of representative results are shown below. They show the performance of the scheduling algorithm for a mixture of traffic sources.

The last two columns in Table 1 show the achieved performance. We have compared the actual achieved bandwidth against what can be expected if we had an ideal scheduler ; the ideal number is provided in braces. For a 512kb/s link, the aggregate guaranteed throughput is $(32+48+8+40+8+40+20+30) = 216$ kb/s. However, if we adjust for the fact that two of the terminals are relatively less efficient, the actual aggregate comes to 240kb/s. This leaves a surplus of 276 kb/s, which have to be distributed according to the THP. The theoretical target bandwidth for a given transaction is given by:

$$B_{theor}[i] = \min(B_{source}[i], MGB[i] + surplus * THP[i] * PCI[i])$$

The charts below show a snapshot sample of the operation of the scheduler. We note certain things. One is that rate-adaptive traffic tends to be more aggressive than delay adaptive traffic and managed to get more of a share of the bandwidth. Delay adaptive traffic on the other hand tends to vary more (even in steady state). The other clear result is the amount of service acquired by the non-adaptive variable bit-rate traffic, simply by dint of pumping more data into the system.

Table 1. Mixture of transactions used for testing the scheduler; source and results.

No.	Title	QoS		Source characteristics				Performance	
		MGB	TH P	Type	Load (kb/s)	Period (ms)	TDS	Tput (kb/s)	Delay (s)
1	32kb/s	32	0.02	CBR	32	80	Hi	32 (32)	0
2	48kb/s	48	0.02	CBR	48	160	Hi	48 (48)	0.08
3	Rate-halving, good link	8	0.35	Rate adaptive	108		Lo	108 (100)	0.08
4	Rate-halving, med. Link	40	0.03	Rate adaptive	108		Lo	70 (48)	0.48
5	Delay adaptive good link	8	0.35	Delay adaptive	108		Med	81 (100)	0.32
6	Delay adaptivemed. Link	40	0.03	Delay adaptive	108		Med	48 (48)	3.0s
7	VBR1	20	0.2	VBR	150	160	Lo	64 (20)	-
8	VBR2	30	0.2	VBR	15	160	Lo	15 (15)	0.04

Looking at the per-packet delay characteristics, we see that all the traffic sources achieve equilibrium, other than the VBR traffic which has more demand for resources than will be allocated to it by the scheduler. This leads to a runaway increase in the per-packet delay, and eventual failure. On the other hand, VBR traffic which is transmitting data at its guaranteed minimum rate also achieves stable delay performance, though it is non-adaptive to traffic conditions.

4 Conclusion

In this paper, we have presented a modification of the standard GPS class of scheduling algorithm. The algorithm allows us to handle a special situation where the transmission is not fully controllable by the scheduler. By using two different QoS attributes, we have shown that our system can handle multiple traffic sources of different types, thus being suitable for mixed (voice, video, user data) systems.

The scheduler that we present here is a very special case, but it also brings to light the requirements of systems which do not fully fall into the ‘packet by packet’ scheduling paradigm. For example, a system which schedules at fixed time boundaries and has to deal with unpredictable but bounded capacity variations i.e. insertion of control traffic or interference related issues will require modifications in the scheduler architecture to take care of these considerations. Further work would include use of wireless scheduling algorithms and handling location dependent packet dropping.

Further work involves handling ‘bounded external unfairness’, due to, for example, the external burst-building and transmission algorithm using concepts of wireless scheduling.

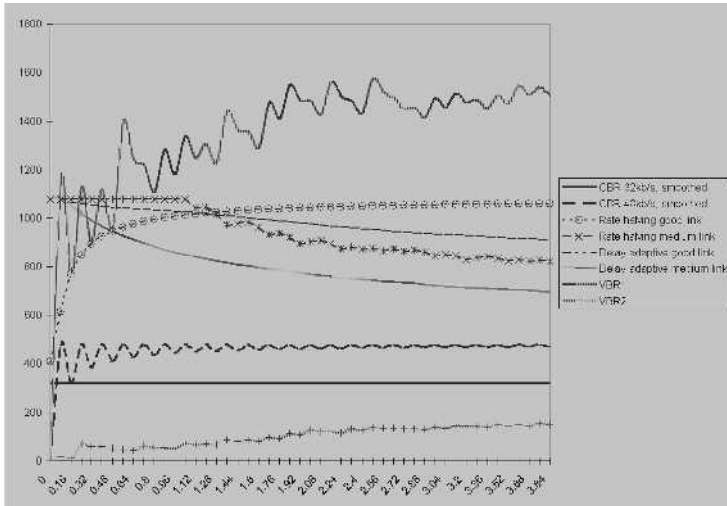


Fig. 1. Input loading for each transaction in bytes per second

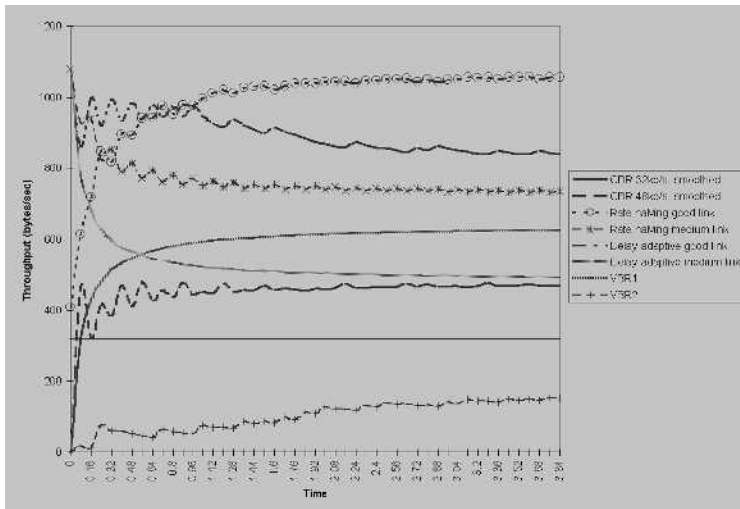


Fig. 2. Service rate for individual connections in bytes/sec

References

- [1] T.S.Eugene Ng, Ion Stoica, Hui Zhang, “Packet Fair Queueing Algorithms for Wireless Networks with Location-Dependent Errors”, Proc. of the INFOCOM, 1998

- [2] Abhay Parekh and R. Gallagher, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Network: The Single Node Case," *IEEE/ACM Transactions on Networking*, April 1994
- [3] Zhi-Li Zang, John Towsley, Jim Kurose, "Statistical analysis of the Generalized Processor Sharing Scheduling Discipline", *Proceedings of the ACM Sigcomm*, 1994
- [4] Jamaloddin Golestani, "A Self Clocked Fair Queueing Scheme for Broadband applications", *IEEE Infocom* 1994
- [5] Jon C.R.Bennett, Hui Zhang, "WF²Q: Worst Case Fair, Weighted Fair Queueing", *Proceedings of the IEEE Infocom*, March, 1996
- [6] T.S.Eugene Ng, Ion Stoica, Hui Zhang, "A Hierarchical Fair Service Curve Algorithm for Link-sharing, Realtime and Priority services", *Proceedings of the ACM Sigcomm*, August 1997
- [7] Scott Shenker, "Making Greed work in Networks: A Game theoretic analysis of Switch Service Disciplines", *ACM/IEEE Transactions on Networking*, 1995
- [8] Scott Shenker, "Fundamental Design Issues for the Future Internet", *IEEE Journal on Selected Areas of Communication*, 13(7), 1992

A Power-Allocation-Combined Scheduling Algorithm for CDMA-Based High-Rate Packet Data Systems

Insoo Koo¹, Jens Zander², and Kiseon Kim¹

¹ Dept. of Infor. and Comm., Kwang-Ju Institute of Science and Technology,
1 Oryong-dong, Puk-gu, Kwangju, 500-712, Korea

² Radio Communication Systems, S3
Isafjordsgatan 30B, KISTA, Stockholm, Sweden.

Abstract. In CDMA-based packet data systems such as HDR and HSDPA which are designed to support high rate services, BSs transmit data packets with a maximum power based on time multiplexing mode such that only one user can be serviced at a time. In this paper, we propose a power-allocation-combined scheduling algorithm for HDR-like systems in which we adopt a code division multiplexing (CDM) transmission method in the downlink common channel in order to utilize channel orthogonality such that we can serve more than one user at a time slot specially when there exist remaining resources after serving the firstly selected user by the scheduler. Simulation results show that the proposed scheme outperforms the conventional scheme as the traffic load increases.

1 Introduction

CDMA-based high-rate packet data systems, called as the 3.5th generation CDMA systems beyond IMT-2000, such as HDR and HSDPA adopt common shared channels in the forward link in order to provide a high bit rate packet data service and an improved throughput. For example, HDR systems adopt the forward packet data channel (FPDCH)[1] while HSDPA systems use the high speed downlink shared channel (HS-DSCH)[2]. These common shared channels are capable of supporting high bit rate by employing adaptive modulation and coding with Hybrid ARQ, turbo codes and transmit diversity. One of the most distinct features of these common shared channels however is to adopt rate adaptation and to service multiple packet data users based on time multiplexing mode. In addition, a mix of services with different requirements is expected to be serviced in the context of next generation CDMA systems. In order to support the quality of various services through the common shared channels, efficient MAC protocols are needed. More specially, it is expected that a scheduling algorithm among MAC protocols plays an important role in the common shared channels since it controls the allocation of the shared resources among users and to a large extent determines the overall behavior of the system.

Recently, many works have been done regarding scheduling algorithm for common shared channels in order to increase total throughput and guarantee

QoS requirements of users [3,5]. A proportionally fair scheduling algorithm proposed in [3] takes advantage of short-term channel variations while at the same time maintaining almost the same long-term throughput among all users such that it can increase system throughput and achieve some degree of fairness among users. As a modification of the scheduling algorithm [5], Kim et. al. suggested an algorithm to provide priority for users by introducing weighting factor. The scheduling algorithms[3,5], however, basically exploit time-varying channel conditions to make a scheduling decision. That is, the previous works only utilize the data-rates requested by mobile station (MS) based on the channel information while ignoring the aspects of queuing by assuming that all buffers are always full. In practice, there are not always data packets to send in the each user's queue such that for certain time the demanded data rate to transmit current packets in the queue can be less than the feasible data-rate requested by MSs. In this situation, we can waste system resource by allowing more transmission power than the demanded one if we assign the resource only based on the feasible data-rates requested by MSs.

With the motivation of this idea, we in this paper suggest a scheduling algorithm which utilizes both queueing information of base station (BS) and channel information of each MS, and further uses code division multiplexing (CDM) in the downlink common shared channel. In the proposed scheme, if the demanded data rate of the user who is selected by scheduler is less than the data-rate requested by MS (i.e., in aspects of transmission power, it corresponds that the demanded power is less than the maximum power level of BS), then scheduler enters into the CDM mode where BS calculates a proper power level to send data packet of the selected user and further selects another user who can utilize the remaining power most efficiently. Finally, BS will send data packets through CDM transmission and MSs receive their packets from the serving BS.

The rest of the paper is organized as follows. In Section II, we review the operation of conventional HDR-like systems. In Section III, we present the proposed algorithm where scheduling algorithm and power allocation are combined through CDM transmission in downlink common shared channel. In Section IV, we show simulation results, and finally draw conclusions in Section V.

2 Conventional HDR-Like Systems

In this paper, we consider the common shared channel of HDR system, namely forward packet data channel (FPDCH). It consists of a single data channel that is divided into 1.25ms time slots. Two pilot bursts are inserted into each time slot to aid synchronization, signal to interference plus noise ratio estimation and coherent demodulation. Control channels and user payload are time-multiplexed onto the forward link.

Pilot burst is transmitted by a constant power from each BS and aids in synchronization and SIR prediction at MSs. Then, each MS measures the pilot-signal SIR, and determines the feasible data-rate that can be supported in the current channel state, based on the quality of the received signals. Channel

Table 1. DRC value under consideration.

RAI Rate (kbps)	Packet Lengths (Bits per Encoder Packet)	Number of slots
9.6	384 (1 IP Packet)	32
19.2	384 (1 IP Packet)	16
38.4	384 (1 IP Packet)	8
76.8	768 (2 IP Packet)	8
153.6	768 (2 IP Packet)	4
307.2	1536 (4 IP Packet)	4
614.4	1536 (4 IP Packet)	2
1228.8	1536 (4 IP Packet)	1
921.6	2304 (6 IP Packet)	2
1843.2	2304 (6 IP Packet)	1
2457.6	3072 (8 IP Packet)	1

information such as feasible data-rate and the received SIR level is also reported to BS by each MS through the data request channel (DRC), one of the reverse link channels. For example, Table.1 shows the data-rate of traffic channel under consideration.

On the side of BS, scheduler makes a decision on which user to choose for the next transmission slot by utilizing channel information of each MS as well as queue information of BS. And then, BS transmits data packets to the selected user with the requested data rate, one of following data-rates; 9.6kbps, 19.2kbps, 38.4kbps, 76.8kbps, 153.6kbps, 307.2kbps, 614.4kbps, 921.6kbps, 1228.8kbps, 1843.2kbps, 2457.6kbps. Fig.1 shows overall schematic structure of the system being considered.

3 A Power-Allocation-Combined Scheduling Algorithm

In order that a scheduler make an appropriate decision on which user to be chosen for the next transmission slot, information on the channel of MSs as well as on the queues of each user at BS is needed as like Fig.1. The channel information of each MS is feasible data-rate, which can be supported by each MS under the current channel condition, while the queue information of BS is the number of packets to send and packet delay of respective queue for each user. By using information about individual data streams, together with information about channel characteristics of different MSs, the scheduler makes a plan the transmission so that the system performance will be maximized while QoS requirements of each user are satisfied. In the previous works[3,5], the scheduler only utilized the feasible data-rates requested by MSs while ignoring the aspects

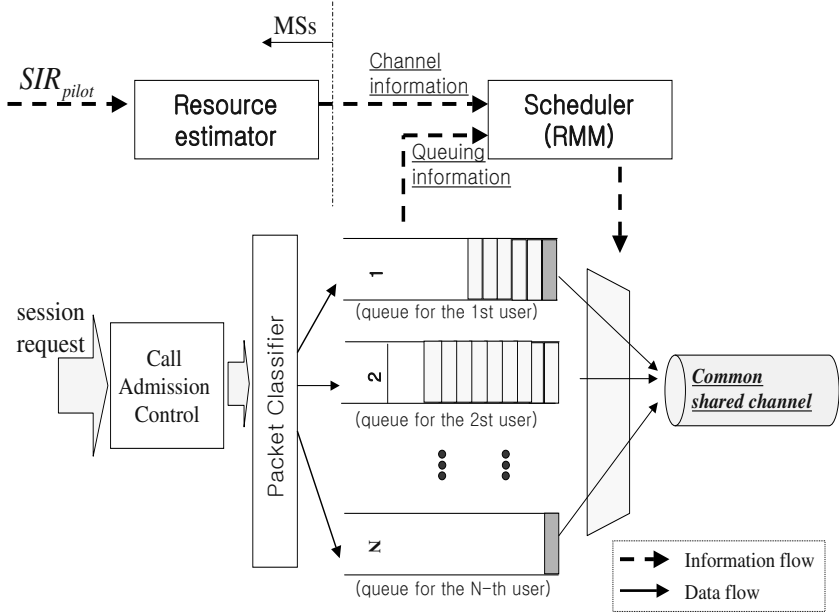


Fig. 1. Overall structure of HDR-like systems.

of queuing by assuming that all buffers are always full. However, this approach can waste system resource by assigning more power than the needed one specially when the demanded data rate is less than the feasible data rate. In this section, we present a scheduling algorithm which utilizes both the queueing information in BS and the channel information of each MS, and further uses code division multiplexing (CDM) in the downlink common shared channel.

3.1 Feasible Data Rate

Mobile stations measure the SIR of the pilot signal, P_{pilot} transmitted from BS. Since the pilot signals are transmitted by the same transmission power at each BS, the SIR of the pilot signal, SIR_{pilot} from the j -th BS at the i -th MS can be expressed as

$$SIR_{pilot}(i) = \frac{P_{pilot} \cdot L(d_{i,j})}{\sum_{k=1, k \neq j}^{K_b} P_{pilot} \cdot L(d_{i,k})} = \frac{L(d_{i,j})}{\sum_{k=1, k \neq j}^{K_b} L(d_{i,k})} \quad (1)$$

where $L(d_{i,j})$ is the path loss from the j -th BS to the i -th MS, and K_b is the number of BSs in the service area. Here we assume that the path loss of forward link between the i -th MS and the j -th BS, $L(d_{i,j})$ is characterized by $d_{i,j}^{-\alpha} \cdot 10^{\epsilon/10}$

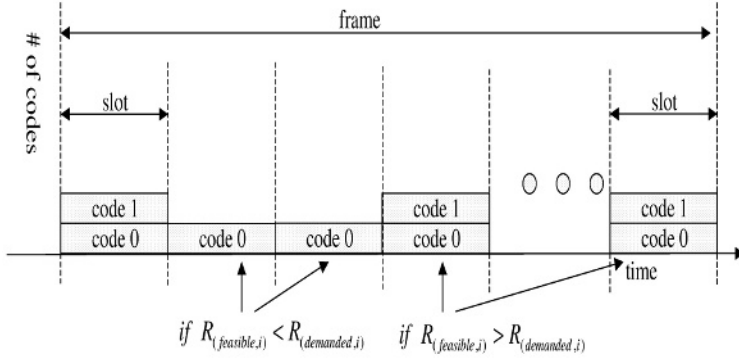


Fig. 2. Code division multiplexing transmission in the common shared channel.

where l is the pass-loss exponent, ξ is a gaussian distributed random variable with zero mean and station deviation σ to consider the effect of shadow fading. Typically, σ takes the value of 6 to 10 [dB] for signals from adjacent BSs and that of 2 to 2.5 [dB] for signals from home BS.

Since HDR-like systems basically adopt rate control scheme, the assigned data rate to a user is dependent on the received SIR. The highest data rate is assigned to each user as long as the received E_b/N_o is larger than the required E_b/N_o for target frame error rate (FER). Subsequently, MSs can estimate the feasible data rates after calculating the received E_b/N_o by measuring the pilot signal such that the feasible data rate of user i that can be achievable with the maximum transmission power of BS, $R_{feasible,i}$, can be given as

$$R_{feasible,i} \leq W \cdot \frac{(SIR)_{pilot}}{(E_b/N_o)_{req}} \quad (2)$$

where W denotes the spreading bandwidth or chip rate, and $(E_b/N_o)_{req}$ is the required E_b/N_o to keep the frame error rate below than target value.

3.2 Demanded Data Rate

The demanded data rate of each MS, $R_{demanded}$ is dependent on the amount of the packets to send in each respective queue. For example, if there are 4 packets with packet length of 384 bits in the queue of the i -th user, then the demanded data rate of the i -th user is 1.2288M *bps* when one time slot duration is 1.25ms. In a certain time, the demanded data rate could be less than the feasible data rate according to the number of packets in the respective queues.

3.3 Code Division Multiplexing Transmission

In the proposed scheme, if the demanded data rate of the user who is selected by scheduler is less than the data-rate requested by MS (i.e., in aspects of the power,

it corresponds that the demanded power is less than the maximum transmission power of BS), then BS calculates a proper power level to send data packets of the selected user successfully and the remaining power will be assigned to another user who can utilize the remaining power most efficiently. Finally, BS will send data packets by CDM and MSs receive their packets from the serving BS. Fig.2 shows an example of code division multiplexing transmission in the common shared channel. The main procedure of the proposed scheme also can be described as following:

- With the information on both the channel of each MS and the queue of BS, the scheduler selects the user, \hat{i} for the next transmission who can maximize the objective function of scheduler, δ_i . That is, $\hat{i} = \arg\{max\{\delta_i\}\}$. The objective function, δ_i can be various according to the scheduling algorithm employed in BS.
- If the feasible data-rate of the selected user \hat{i} is larger than the demanded data-rate, then BS transmits data packets of the user \hat{i} with the maximum power as like the operation of conventional HDR-like systems. Otherwise, the scheduler enters into the CDM mode in which BS calculates the power fraction of user \hat{i} and the left power $(1 - \phi(\hat{i}))$ will be allocated to another user who can utilize the remaining power most efficiently. More detailed description for the CDM mode is also as following:
- In the CDM mode, we calculate the proper power level to send the data packets of user \hat{i} successfully. When the radio channel is static during a frame, E_b/N_o of the packet that will be received in the following time slot from the BS j at the \hat{i} -th MS, $E_b/N_o(\hat{i}, j)$ can be expressed by

$$E_b/N_o(\hat{i}, j) = \frac{\phi_A(\hat{i}) \frac{P_{total}(j)}{L(d_{i,j})} \frac{W}{\min(R_{feasible,\hat{i}}, R_{demanded,\hat{i}})}}{\left((1 - \bar{F}_o)(1 - \phi_A(\hat{i})) \frac{P_{total}(j)}{L(d_{i,j})} + \sum_{k=1, k \neq j}^{K_b} \frac{P_{total}(k)}{L(d_{i,k})}\right)} \quad (3)$$

where $P_{total}(j)$ is the total transmission power at the j -th BS, $\phi_A(\hat{i})$ is a ratio of transmission power for the \hat{i} -th MS to the total transmission power $P_{total}(j)$, and \bar{F}_o is an average orthogonality factor defined as the fraction of total received power that will be experienced as intra-cell interference due to multi-path propagation. The \bar{F}_o is 1 for perfect orthogonality and 0 for non-orthogonality.

Here we can assume that $P_{total}(j)$ is the same for all BSs. Then, $E_b/N_o(\hat{i}, j)$ can be estimated by using the following equation:

$$E_b/N_o(\hat{i}, j) = \frac{W}{\min(R_{feasible,\hat{i}}, R_{demanded,\hat{i}})} \cdot \frac{\phi_A(\hat{i}) SIR_{pilot}(\hat{i})}{(1 - \bar{F}_o)(1 - \phi_A(\hat{i})) SIR_{pilot}(\hat{i}) + 1} \quad (4)$$

Assuming that data packet is successfully received at MS when its received $E_b/N_o(\hat{i})$ is greater than the required E_b/N_o , $(E_b/N_o)_{req}$, we can calculate the condition of $\phi_A(\hat{i})$ in order to correctly receive packets at MS such that

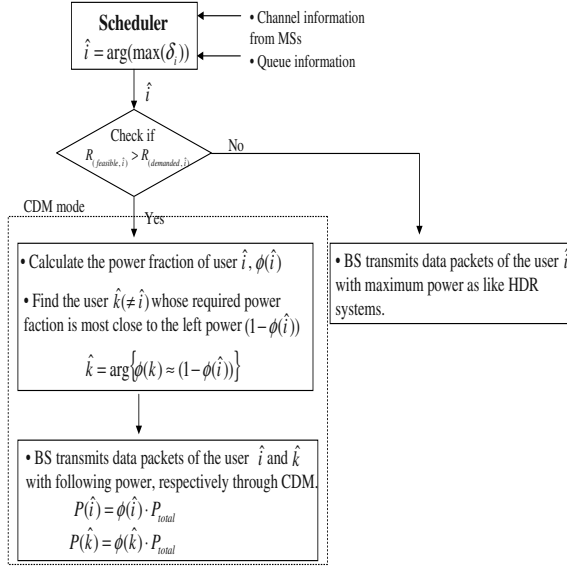


Fig. 3. The flow-chart of the proposed scheme.

$$\phi_A(\hat{i}) \geq \frac{(E_b/N_o)_{req}(SIR_{pilot}(\hat{i})(1 - \bar{F}_o) + 1)}{SIR_{pilot}(\hat{i})((E_b/N_o)_{req}(1 - \bar{F}_o) + \frac{W}{\min(R_{feasible,\hat{i}}, R_{demanded,\hat{i}})}} \quad (5)$$

Hence, the required transmission power $P_{req}(\hat{i})$ for the \hat{i} -th MS at the j -th BS can be estimated by using the following equation:

$$P_{req}(\hat{i}) = \phi_A(\hat{i}) \cdot P_{total}(j) \quad (6)$$

- After that, we find the user \hat{k} ($\hat{k} \neq \hat{i}$) who can fully utilize the remaining power of the j -th BS, $(1 - \phi_A(\hat{i})) \cdot P_{total}(j)$. That is, the scheduler selects the user \hat{k} whose required power fraction is most close to the power fraction $(1 - \phi_A(\hat{i}))$ such that

$$\hat{k} = \arg\{\phi(k) \approx (1 - \phi_A(\hat{i}))\} \quad (7)$$

- Finally, BS transmits the data packets of user \hat{i} and \hat{k} with the following power, respectively.

$$P(\hat{i}) = \phi_A(\hat{i}) \cdot P_{total}(j) \quad (8)$$

$$P(\hat{k}) = \phi_A(\hat{k}) \cdot P_{total}(j) \quad (9)$$

Fig.3 finally shows the flow-chart of the proposed scheme.

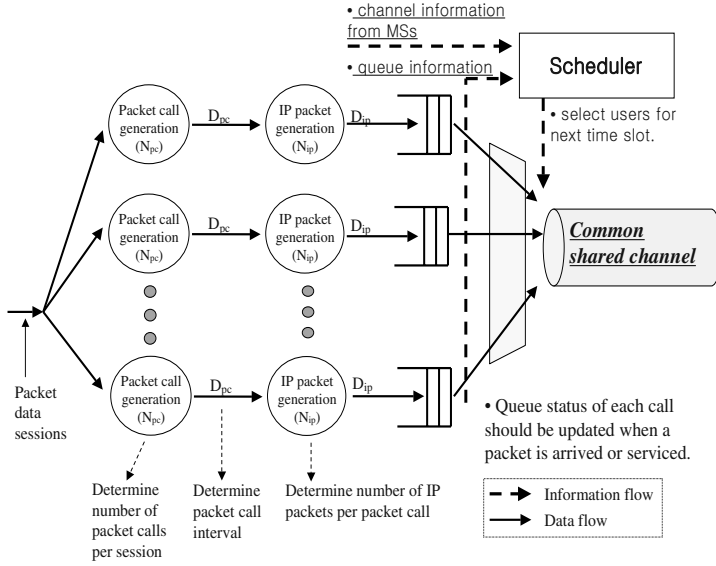


Fig. 4. Schematic of overall simulation environment.

4 Simulation and Results

Fig. 4 shows the schematic of overall simulation environment. As like the figure, multiple sessions are given to the FPDCH. When IP packets of a packet call in each session arrive at a BS, these data packets are buffered in respective queues at BS. According to the scheduling algorithm employed in BS, the service turn for each user (or session) is determined and the data packets are packaged as “encoder packet” according to the service rate, after that they are transmitted through the FPDCH. The receiver of each session receives the signals from the FPDCH, and picks up the encoder packet for himself by detecting the preamble subchannel of the FPDCH. Here we consider the following simulation conditions; • the number of connected sessions are fixed 20, • the buffer size of each queue for each session is 100, • the maximum delay requirement is given 0.1 sec, • the average inter-arrival time of packet calls is 15 msec, • the average number of IP packets per a packet call varies in order to change the traffic load.

Fig.5 shows the delay-outage probability according to the traffic load, which is defined as the probability that the delay experienced by an IP packet exceeds the maximum delay requirement. Here, we use the max-rate scheduling algorithm as a reference to investigate the gain of the proposed scheme. However it is noteworthy that any kind of scheduling algorithms previously proposed in other literature can be combined with the proposed scheme. In the case of the max rate algorithm without information of queuing, we only utilize the channel information such that the scheduler selects the user $j = \arg\{max\{R_{feasible,i}\}\}$

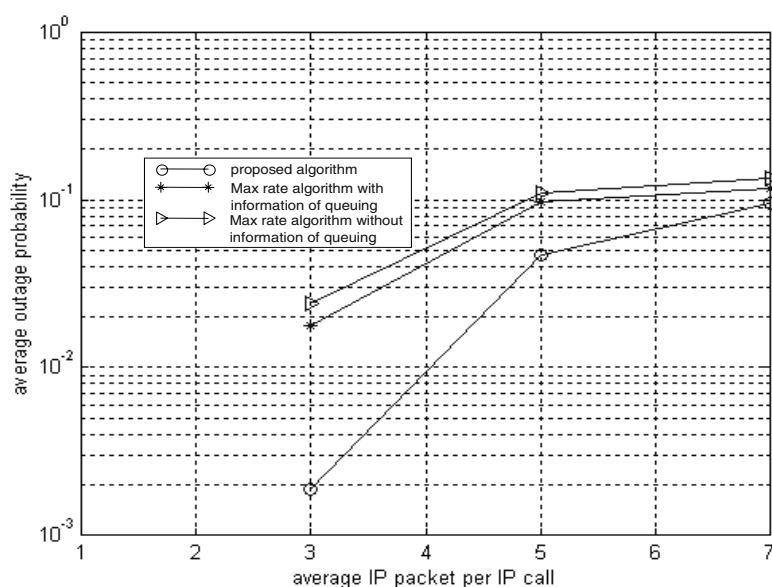


Fig. 5. Average delay-outage probability according to traffic load.

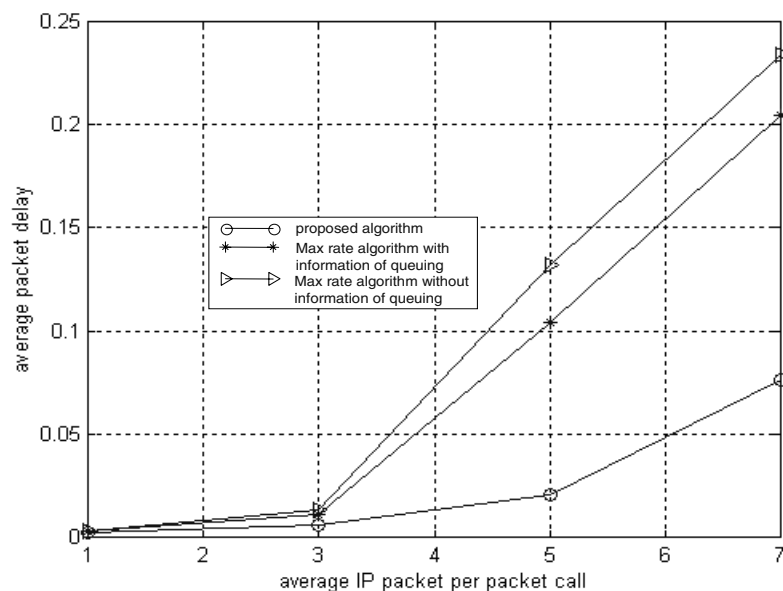


Fig. 6. Average packet delay according to traffic load.

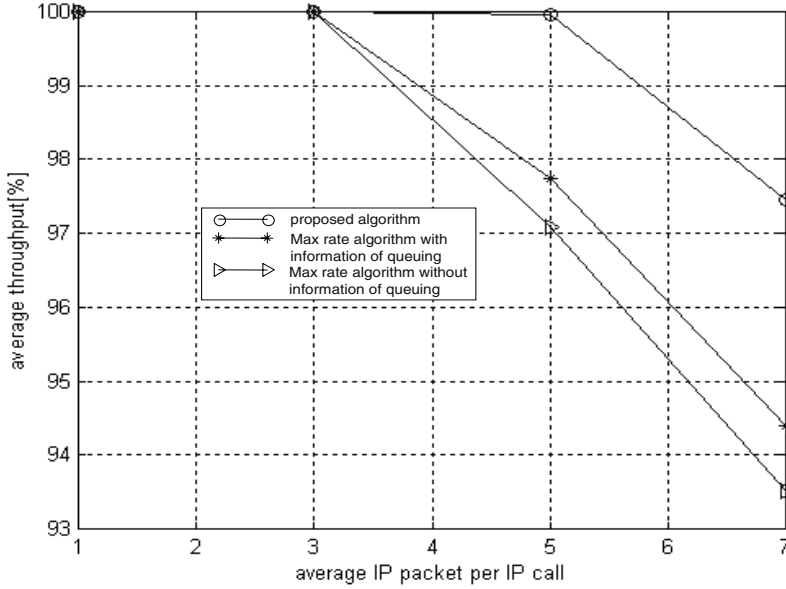


Fig. 7. Average throughput according to traffic load.

for the next slot transmission. On the other hand, in the case of the max rate algorithm with information of queuing, the scheduler selects the user $j = \arg\{\max\{\min(R_{feasible,i}, R_{demanded,i})\}\}$ by using the channel information as well as queuing information. In the case of the proposed scheme, we further utilizes CDM mode in addition to the max rate algorithm with information of queuing. From Fig.5, we know that the proposed scheme outperforms the max rate algorithm without or with information of queuing. Fig.5 also shows that the max rate algorithm with information of queuing slightly outperforms the max rate algorithm without information of queuing in the aspects of average outage probability.

Fig.6 shows average packet delay as a function of traffic load. The average packet delay is defined as service time that an IP packet experiences from the time to arrive at BS to the time when the packet is successfully transmitted to the user. From Fig.6, we know that the proposed scheme more outperforms the other two schemes as the traffic load increases.

Fig.7 shows average throughput as a function of traffic load. Here, the average throughput is defined as the ratio of the number of successfully transmitted IP packets to the total number of generated IP packets. From Fig.7, we also know that the proposed scheme more outperforms the other two schemes as the traffic load increases.

Even though we have showed the simulation results when the max rate algorithm is utilized as the basic scheduling algorithm, we can expect that we will get similar performance gains for any kind of scheduling algorithms when the proposed scheme is combined with it.

5 Conclusions

In this paper, we have proposed an power-allocation-combined scheduling algorithm for HDR-like systems where we adopt code division multiplexing (CDM) transmission method in the downlink common shared channel in order to utilize channel orthogonality such that we can serve more than one user at a time slot specially when there exist remaining resources after serving the first user selected by the scheduler. Simulation results showed that the proposed scheme outperforms the conventional schemes as the traffic load increases.

Acknowledgement. This work was supported by Korea Science & Engineering Foundation (KOSEF) through the UFON research center at K-JIST. Insoo Koo particularly was supported by grant No. *R08 – 2003 – 000 – 10122 – 0* from the Basic Research Program of the KOSEF.

References

1. 3GPP2 C.S0024 “CDMA2000 high rate packet data air interface specification,” Version 2.0, Oct. 27, 2000.
2. Motorola, etc., “Proposed 1xTREME physical layer delta specification,” Oct. 24, 2000.
3. M. Andrews, et al., “Providing quality of service over a shared wireless link,” IEEE Comm. Magazine, pp. 150–154, Feb. 2001.
4. A. Jalali, R. Padovani and R. Pankaj, “Data throughput of CDMA-HDR a high efficiency-high data rate personal communication wireless system,” Proc. of VTC (spring), vol.3, pp. 1854–1858, 2000.
5. K. Kim, H. Kim and Y. Han, “A proportionally fair scheduling algorithm with QoS and priority in 1xEV-DO,” Proc. of PIMRC, pp. 2239–2243, 2002.

Analyzing the Performance of Data Users in Packet Switched Wireless Systems with Prioritized Voice Traffic

Roshni Srinivasan and John Baras

University of Maryland, College Park MD 20742, USA
{roshni, baras}@eng.umd.edu

Abstract. The integration of wireless telephony and data services in 3G and 4G wireless systems that use packet-switched air interfaces poses new challenges in the management of network resources. Although highly compressed voice traffic is given priority over data traffic, scheduling algorithms which exploit multiuser diversity have been shown to significantly improve the data throughput. In this paper, we quantify the effect of prioritized voice traffic on the performance of data users in the system using a mix of analysis and simulation. We analytically characterize the scheduled rate, delay and packet service times for data in the presence of prioritized voice traffic by using a general scheduling metric that incorporates a measure of the user's channel quality in addition to a delay constraint. The results provide important tools for cellular network operators to evaluate system performance and provision resources for traffic with varying Quality of Service(QoS) requirements.

1 Introduction

The support of real-time services in packet-switched 3G and 4G cellular wireless systems is currently a topic of active research. In particular, there is significant interest in the integration of voice over IP (VoIP) and data traffic. In 1xEV-DO [1,2], if 20 ms speech frames generated by VoIP codecs such as G.729 are to be delivered with minimal delay, 24 voice users can be accommodated, assuming a voice activity factor of 0.5. A higher number of voice users can be served by either using higher compression rates, or by tolerating a larger amount of scheduling delay, both of which can adversely affect voice quality. The simplest technique to support delay-sensitive traffic such as packet voice in a packet-switched cellular data system is to strictly prioritize it over data traffic. 1xEV-DO, for instance, supports QoS by prioritizing delay-sensitive data in the wireline backhaul network as well as over the airlink [3]. The residual bandwidth (time slots) available for data applications can be utilized most efficiently by exploiting multiuser diversity techniques [4,5]. This form of diversity exploits independent fading in a multiuser environment by opportunistically scheduling users at favorable channel instants. However, unfair resource allocation and variability in scheduled rate and delay are natural consequences of such algorithms [6,7].

This paper focuses on the impact of supporting VoIP services in a time-slotted packet-switched air interface. The system is assumed to use strict prioritization for voice, while data packets are opportunistically scheduled subject to delay constraints. As in 1xEV-DO, mobile users are assumed to report the maximum sustainable downlink rate, $R(t)$ to the base station via a dedicated channel on the uplink in order to support opportunistic scheduling. We consider a general scheduling metric, originally introduced in [8], that combines channel state with delay constraints in the form

$$m(t) = R(t) + \alpha \frac{v(t)}{N_d} = R(t) + \alpha V(t), \quad (1)$$

where $v(t)$ is the scheduling delay for a waiting packet and α is a configurable control weight that allows control of delay at the expense of multiuser diversity gain. While the scheduler is not designed to optimize either throughput or delay constraints, the metric elegantly captures the trade-off between multiuser diversity gain and delay through the control parameter, α . Also, this metric lends itself well to statistical analysis as well as implementation. In order to ensure that the delay does not dominate the scheduler metric as the number of data users N_d increases, we normalize $v(t)$ by N_d . Hereafter, we refer to $V(t)$, the normalized scheduling delay, as *vacation time*.

The original contribution of this paper is an analytical characterization of the effect of prioritized voice users on data users in a cellular wireless system with delay constrained opportunistic scheduling. We quantify the resulting delay and compute the packet service times for data users as a function of the number of voice users in the system when all users have fully loaded queues and the resources of the system are completely utilized. The outline of the paper is as follows. Section 2 discusses related work in this area of research. In the analysis presented in Section 3, we derive expressions for the distribution of scheduling jitter and scheduled rate, which in turn allow computations of the distributions of packet service times. Section 4 contains details of the system model. Simulation results that validate our analysis are presented in Section 5. Finally, we highlight the main ideas in this work and summarize our results in Section 6.

2 Related Work

Wireless networks that were designed to support circuit-switched voice traffic are now migrating to packet-switched networks that support data applications as well [9]. Although QoS support for real-time traffic in wireline networks has been well-studied in the literature, the time-varying wireless channel adds a new dimension to the problem. QoS provisioning for wireless systems that incorporated channel-state dependent scheduling algorithms are outlined in [10] and in [11]. In the former, the authors propose the Modified - Largest Weighted Delay First (M-LWDF) rule to optimally provide QoS guarantees in terms of predefined guarantees for the probability of loss and minimum long-term throughput for each user. The exponential rule presented in the latter provides an effective

way of realizing multiuser diversity gains with a delay constraint. In recent work, the authors in [12] introduce the concept of effective capacity to explicitly guarantee QoS.

Packet-switched wireless networks that primarily support voice calls through prioritization of voice traffic can share unused voice bandwidth among data applications. However, the number of such data users that can be supported is limited by the user experience in terms of delay and throughput. In order to quantify the throughput and delay experienced by data users in the presence of voice traffic, we model the scheduler as a dynamical system. We analyze the performance of the data users at the fixed point as a function of the number of voice users in the system. The results, which are validated by simulations of an actual system provide a network operator useful tools in evaluating system performance for a given mix of voice and data traffic.

3 Delay and Throughput Analysis for Data Users

In this section, we present an analytical framework to evaluate the throughput and delay of data traffic with prioritized voice traffic. This framework was originally developed in [8] to analyze the trade-off between throughput and delay in opportunistic schedulers. When users have different channel statistics, the scheduler metric in 1 can be modified to ensure resource fairness as follows:

$$m_i(t) = (R_i(t) + \gamma_i) + \alpha V_i(t) \quad (2)$$

where γ_i can be chosen optimally [13] to maximize the total scheduled rate while ensuring temporal fairness without delay constraints ($\alpha = 0$). In order to better understand the interaction of voice and data users, we assume identical channel statistics for all data users. As described in Section 1, we denote the rate requested by data user i at time t by $R_i(t)$. Let $\mathcal{R} = \{r_0, r_1, \dots, r_{max}\}$ denote the finite set of rates requested by the users. This set is assumed to have a probability distribution $f_{R_i}(r) = f_R(r) = P(R = r), r \in \mathcal{R}, \forall i$. The delay experienced by data user, i since it was previously scheduled is $v_i(t)$, with $V_i(t) = v_i(t)/N_d$ representing the normalized vacation time.

In every time slot, voice users are served with the highest priority on a first-come first-served (FCFS) basis. If there is no waiting voice packet, the base station transmits to the user with the highest metric as computed from equation 1. In the event that more than one data user has the highest metric, a data user is picked with uniform probability from among the users with the highest metric in order to break the tie. The complexity of the joint state-space resulting from the combinations of scheduling delays and requested rates makes the analysis of the scheduler using Markov models intractable. In [8], we define a permutation of the data user space in which the users are rank-ordered in every slot according to the delay they have experienced since they were last scheduled. Let $\mathbf{U}(t)$ denote the rank-ordering of the N_d data users at time t :

$$\mathbf{U}(t) = \{u_0(t), u_1(t), \dots, u_{N_d-1}(t)\} \quad (3)$$

where $u_i(t) \in [0, 1, \dots, N_d - 1]$. In this space, $u_i(t)$ denotes the original index of the data user who is ranked in the i^{th} position at time t . By definition,

$$V_{u_0}(t) \leq V_{u_1}(t) \leq \dots \leq V_{u_{N_d-1}}(t) \quad (4)$$

where $V_{u_i}(t)$ is the vacation time seen by the data user who is ranked in position i at time t . Naturally, since $u_0(t)$ is the index of the data user who was scheduled in the current slot, $V_{u_0}(t) = 1/N_d$. At time t , if no voice packet is available for transmission, the scheduler selects a data user whose rank-ordered index is given by $S^*(t)$ where

$$S^*(t) = \underset{i}{\operatorname{argmax}} m_{u_i}(t) \quad (5)$$

The scheduling decision in one slot affects the rank-ordering at the beginning of the next slot. At time $(t+1)$, the scheduled data user, S^* , at time t now moves to position 0. All data users below the rank of S^* increment their rank by one. However, all data users with rank greater than that of S^* do not change their order in any way. Naturally, if a voice packet is scheduled, the rank-ordering of the data users remains invariant. \mathbf{U} evolves over time as:

$$u_i(t+1) = \begin{cases} u_{S^*(t)}, & i = 0 \\ u_i(t), & i = S^*(t) + 1, S^*(t) + 2, \dots, N_d - 1 \\ u_{i-1}(t), & i = 1, 2, \dots, S^*(t) - 1 \end{cases} \quad (6)$$

Similarly, the vacation times seen by the data users change with every packet that is scheduled.

$$V_{u_i}(t+1) = \begin{cases} \frac{1}{N_d}, & i = 0 \\ V_{u_{i-1}}(t) + \frac{1}{N_d}, & 0 < i < S^*(t) \\ V_{u_i}(t) + \frac{1}{N_d}, & i > S^*(t) \end{cases} \quad \text{if data pkt is scheduled} \quad (7)$$

$$V_{u_i}(t+1) = V_{u_i}(t) + \frac{1}{N_d}, 0 \leq i < N_d \quad \text{if voice pkt is scheduled} \quad (8)$$

We define a selection density function, $\pi_{u_i}(t)$ which represents the probability of scheduling the i th rank-ordered data user, u_i at time t .

$$\pi_{u_i}(t) = \Pr(S^*(t) = u_i) \quad (9)$$

with the property, $\sum_{i=0}^{N_d} \pi_{u_i}(t) = 1$. We now analyze the dynamical system consisting of the rank-ordered data user space, the corresponding channel conditions and scheduling delays. An iterative computation of $V_u(t)$ and $\pi_u(t)$ converges to the fixed-point, time-invariant solutions of the dynamical system, i.e., V_u and π_u .

3.1 Computation of the Vacation Function, V_u

The vacation function, V_u characterizes the normalized delay expressed by the data users in the system resulting from a choice of the scheduler metric. In the

analysis that follows, we assume the existence of a selection density function, $\pi_{u_j}(t)$ which represents the probability of scheduling the j th rank-ordered data user, u_j at time t . Observe from equations 6 and 7 that the vacation function at position i in the rank-ordered space is subject to two transforming forces. The first causes its value to increase by $1/N_d$ when either a voice user is scheduled or a data user with a rank less than i is scheduled. If the probability of scheduling a voice user in any slot is given by p_v , then this event occurs with probability $p_v + (1 - p_v)(\sum_{j < i} \pi_{u_j})$. The second transformation causes its value to decrease whenever the rank of the data user scheduled is i or higher. In this event, the value of the vacation-time at position i is replaced by that at position $(i - 1)$, augmented by $1/N_d$. The probability of this event is $(1 - p_v)(\sum_{j \geq i} \pi_{u_j})$. At equilibrium, the function is invariant to these transforming forces. The potential increase can therefore be equated to the potential decrease.

$$\begin{aligned} & \frac{1}{N_d}(p_v + (1 - p_v)(\sum_{j < i} \pi_{u_j})) = \\ & ((1 - p_v)(\sum_{j \geq i} \pi_{u_j}))(V_{u_i}(t) - (V_{u_{i-1}}(t) + \frac{1}{N_d})) \end{aligned} \quad (10)$$

Dropping the dependence on time and applying the boundary condition, $V_{u_0} = \frac{1}{N_d}$, V_{u_i} may be computed recursively as

$$V_{u_i} = V_{u_{i-1}} + \frac{1}{N_d(1 - p_v) \left(1 - \sum_{j < i} \pi_{u_j}\right)} \quad (11)$$

3.2 Computation of the Selection Density Function π_u

The data user with rank-ordered index i has a vacation time of V_{u_i} . If R_{u_i} is its requested rate, its scheduling metric is given by

$$m_{u_i} = R_{u_i} + \alpha V_{u_i} \quad (12)$$

The probability of selecting the i^{th} data user is then given by

$$\begin{aligned} \pi_{u_i} &= P(m_{u_i} > m_{u_j} \quad \forall j \neq i) + P(u_i \text{ selected in tie}) \\ &= P(R_i + \alpha V_{u_i} > R_j + \alpha V_{u_j} \quad \forall j \neq i) + P(u_i \text{ selected in tie}) \end{aligned} \quad (13)$$

The computation of the probability of a data user being selected in the event of a tie is given in Appendix A in [8]. This probability is accounted for in all numerical evaluations of the analytical results in Section 5.1. Since the channel rates are i.i.d. random variables with distribution $f_R(r)$, the first term reduces to

$$P(m_{u_i} > m_{u_j}) = \sum_{r=r_0}^{r_{max}} \left(\prod_{j \neq i} F_{R_{u_j}}(r + \alpha(V_{u_i} - V_{u_j})) \right) f_R(r) \quad (14)$$

3.3 Distributions for Vacation Time and Scheduled Rate

The vacation function and the selection density function as computed in Equations 11 and 13 respectively can be composed to calculate the distribution of vacation time at the scheduling instants. Let \mathcal{V}_{S^*} represent the vacation time seen by the *scheduled* data user. The CDF of \mathcal{V}_{S^*} is given by

$$P[\mathcal{V}_{S^*} \leq V_{u_i}] = \sum_{j=0}^{u_i} \pi_{u_j} \quad (15)$$

Apart from delay, the other quantity determining the performance of data traffic is the rate at which it is scheduled, which is related to the multiuser diversity gain. The pdf of the scheduled rate (which is naturally different from that of the requested rate) may be derived as a function of α as

$$\begin{aligned} f_{R_{S^*}}(r) &= \sum_{i=0}^{N_d-1} Pr(R_{u_i} = r, i\text{th rank-ordered data user is selected}) \\ &= \sum_{i=0}^{N_d-1} f_R(r) \left(\prod_{j \neq i} F_R(r + \alpha(V_{u_i} - V_{u_j})) + Pr(\text{User } i \text{ selected in tie}) \right) \end{aligned} \quad (16)$$

3.4 Packet Service Times

Observe from Figure 1 that the packet service time may be expressed as the sum of scheduling periods and vacation periods. The packet service time, X , is the sum of X_S , the total number of slots required to transmit all the LL segments corresponding to the packet at the head of the queue and X_V , the number of slots where the scheduler goes on vacation. We assume that packets of fixed length, L are served by the base station in the order that they arrive. Furthermore, the packet service times are i.i.d. and independent of the interarrival times. Since

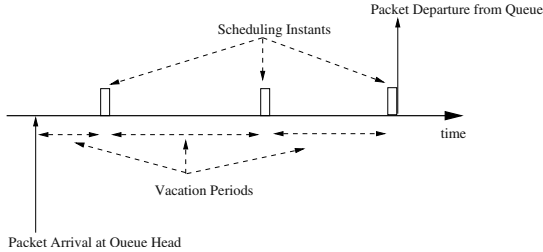


Fig. 1. Illustration of packet service timeline

the scheduling instants are independent of the vacation times, the distribution of service time may be calculated as

$$P(X = n) = \sum_j P(X_V = n - j)P(X_S = j) \quad (17)$$

If the packet completes service in j time slots, then the data user has experienced exactly j vacation periods between the scheduling instants. We represent this sum of j i.i.d. vacations as $V^{(j)}$. Let $R_{S,k}$ represent the scheduled rate corresponding to the k^{th} slot in the transmission of a given packet. The scheduled rates in different slots are assumed to be i.i.d. random variables. The distribution of the packet service times can therefore be computed from the distributions of the scheduled rate and vacation times.

$$P(X = n) = \sum_j P(V^{(j)} = n - j)P(R_S^{(j-1)} < L \leq R_S^{(j)}) \quad (18)$$

4 System Model

In this section, we first outline the model used to describe the wireless channel experienced by mobile users. We then highlight important aspects of the scheduler implementation.

4.1 Wireless Channel Model

Every mobile user is assumed to experience a flat fading channel where the channel response is assumed to be flat for the duration of the slot of K samples. If $x_i(t) \in \mathcal{X}^K$ is the vector of transmitted symbols and $y_i(t) \in \mathcal{X}^K$ is that of the symbols received by user i , then

$$y_i(t) = h_i(t)x(t) + z_i(t) \quad i = 1, 2, \dots, N_d \quad (19)$$

where $h_i(t)$ is the time-varying channel attenuation from the base station to the mobile and $z_i(t)$ is i.i.d., zero mean, additive white Gaussian noise with variance σ_i^2 . Assuming unit-energy signals, the nominal signal-to-noise ratio (SNR) for data user i is $C_{NOM,i} = \frac{1}{\sigma_i^2}$ with the instantaneous SNR for this data user, $C_i(t)$ given by $C_i(t) = \frac{h_i(t)}{\sigma_i^2}$. We assume a Rayleigh SNR distribution and generate the fading coefficients using the well-known Jakes model [15].

We study the performance of 8 data users ($N_d = 8$), all with a nominal SNR of 2.5 dB and a doppler frequency of 10Hz. A scenario with identical channel statistics for all the data users was selected to enable comparison between analysis and simulation. For the case of i.i.d. channel fades, the distribution of the channel rates is chosen to be identical to the marginal distribution obtained with correlated channel fades at the same nominal SNR.

4.2 Air-Interface Model and Scheduler Implementation

We consider a cellular air-interface with a scheduled downlink and a circuit-switched uplink. Voice and data traffic are scheduled at the base station by the scheduler. Packets streams from voice and data users are assigned separate queues by the BS. Fixed length packets of 512 bytes are then segmented into link-layer (LL) segments of 8 bytes for efficient transmission over the air link. At the beginning of each transmission time slot, the scheduler at the BS computes the metric as defined in equation 1 and selects the data user with the highest metric. If there is no waiting voice packet, the BS transmits one or more LL

Table 1. Transmission rate per slot as a function of SNR

SNR (in dB)	Rate (Kb/s)	SNR (in dB)	Rate (Kb/s)
-12.5	38.4	-1.0	614.4
-9.5	76.8	1.3	921.6
-6.5	153.6	3.0	1228.8
-5.7	204.8	7.2	1843.2
-4	307.2	9.5	2457.6

segments to the selected data user over the airlink in every slot, each of which is a fixed duration of 1.667 ms. The number of segments transmitted in a slot depends on the current SNR of the selected data user. Table 1 lists the transmitted rate in link-layer segments as a function of the SNR. The peak rate of 2.45Mbps achievable in this model is similar to that in systems such as 1xEV-DO. When all the LL segments corresponding to the packet at the head of the queue for a particular data user have been transmitted over the airlink, the packet is deemed to be successfully transmitted. Transmission errors can be simulated by probabilistically delaying packet transmission. Since we assume that the channel state is known to a high degree of accuracy, we assume a negligible loss probability. Every user is always assumed to have data in the queue. This ensures that the scheduling metric is the sole criterion for selecting a user.

5 Simulation Results

The effect of admitting prioritized voice users on the throughputs of data users is illustrated in Figure 2, which plots the CDF of the effective throughput experienced by a *packet call*. Our evaluation assumes that the packet call emulates the download of a standard 5KB web page, and is based on standard techniques used in 3GPP [14]. The CDF is obtained by using Monte Carlo simulations to average the effective throughputs of the data call for a number of low-mobility users distributed under different channel conditions in an interference-constrained cell. We plot the packet call throughputs in bits per second with and without voice

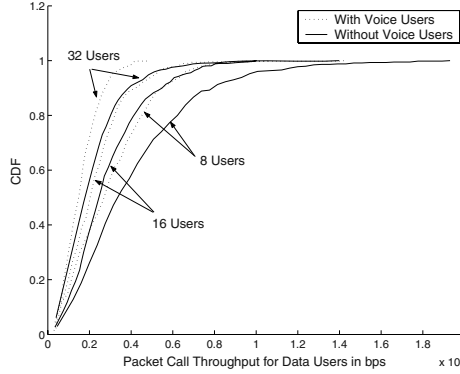


Fig. 2. Impact of 25% voice calls on packet call throughputs of data users

traffic when 8, 16 and 32 data users share 75% of the total voice capacity. Observe from Figure 2 that the performance that the system offers 8 data users in the presence of voice is similar to that obtained by 16 users in the absence of voice. The level of degradation of the data user experience depends not only on the fraction of time slots used by voice users but also on the number of data users sharing the unused voice bandwidth and the type of data traffic.

5.1 Numerical Computation of V_u and π_u

We compare the analytical results in Section 3 with simulations by numerically evaluating the vacation function V_u and the selection density function π_u . Since neither function is known at the outset, we use the following approach to compute the functions iteratively. Let $\pi_u^{(k)}$ and $V_u^{(k)}$ represent the selection density function and the vacation functions estimated in the k^{th} iteration respectively. We start with the Maximum SNR scheduler with $\alpha = 0$ in Equation 1. If the N_d data users have identical channel statistics, this results in a uniform selection function, $\pi_u^{(0)} = \frac{1}{N_d}$. $V_u^{(1)}$ can be therefore be computed using the expression derived in Equation 11. Correspondingly, $\pi_u^{(1)}$ is computed from $V_u^{(1)}$ using the approach outlined in Section 3.2. In subsequent iterations, $V_u^{(k)}$ is computed from $\pi_u^{(k-1)}$, which in turn facilitates computation of $\pi_u^{(k)}$. The convergence of this process has been observed empirically [8].

5.2 Effect of Voice Calls on Vacation Time

The gains from multiuser diversity are maximized by setting $\alpha = 0$ in equation 1. In this case, the selection density function, π_u is uniformly distributed among the data users since the scheduler picks the data user with the best channel without any constraint on delay. Observe from equation 11 that the local slope of the

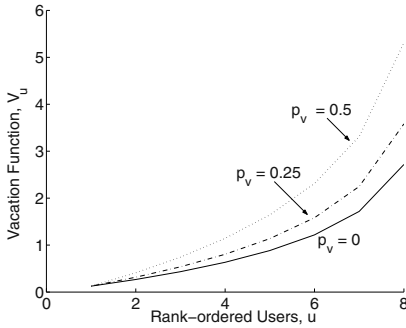


Fig. 3. Impact of voice on vacation function of data users

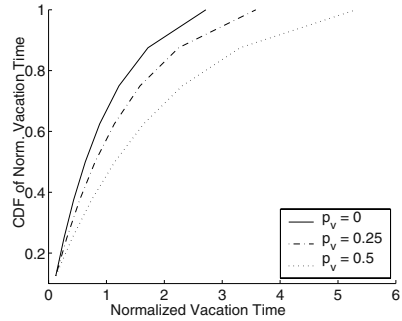


Fig. 4. Impact of voice on CDF of vacation time for data users

vacation function is given by $\frac{1}{N_d(1-p_v)(1-\sum_{j<i} \pi_{u_j})}$. For a fixed number of data users and $\pi_u = 1/N_d$, as the number of voice users increases, p_v increases, thereby causing the local slope of the vacation to increase at every point. As expected, we see in Figure 3 that the increase in slope for the higher rank-ordered data users increases with p_v . We see from Figure 4 that normalized vacation time for data users also increases as they contend for system resources with more voice users.

Multuser diversity gains are maximized for the data traffic by setting $\alpha = 0$ in the metric defined in Equation 1 for our simulations. It is important to note that a metric with α set to some non-zero positive value will only bias the scheduler to favor data users with higher values of V , i.e., lower scheduling delays at the expense of multuser diversity gain. These results are not included in this paper for reasons of compactness. Furthermore, increasing the number of voice users will only cause scheduling delay to further dominate the scheduling metric at the expense of overall system throughput and degrade the performance for best-effort data applications even more.

5.3 Packet Service Time Statistics

We see from equations 11 and 18 that the admittance of voice users naturally causes packet service times and delays for data users to increase. Figure 5 illustrates the CDF of packet service times experienced by data users in the absence of any voice calls. The CDF obtained by numerically evaluating the analytical expression from Section 3.4 is compared with simulated results when the channel rates are i.i.d., and shows very close correspondence. In the case of i.i.d. channel fades, the median packet service time is about 20 slots, while the packet service time at the 90th percentile is about 30 slots. In comparison, Figure 6 illustrates the effect on the packet service times when 50% of the time slots are occupied by voice calls. For i.i.d. fades, the median packet service time is about 35 slots,

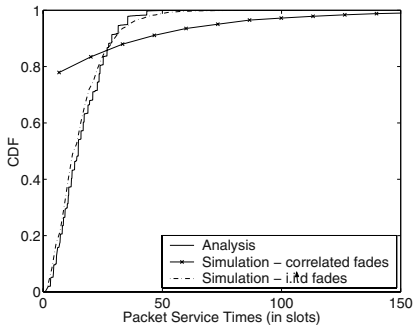


Fig. 5. Packet service time CDF for 8 Data users in the absence of voice

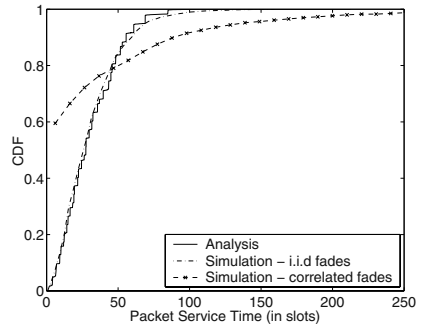


Fig. 6. Packet Service Time CDF for 8 Data Users, 50% Voice Calls

while the packet service time at the 90th percentile increases to 60 slots. The distribution of packet service times in the presence of correlated channel fading, which is obtained through simulation, is also included in these figures. In both cases, the distribution of packet service times for correlated channel fading occupies a much wider dynamic range. This is because a data user may be scheduled repeatedly with lower delays when the channel remains in a good state and higher delays when it remains in a bad state.

6 Conclusions

Cellular wireless systems that traditionally supported voice traffic now see an increasing demand for data services. Packet-switched time-slotted air interfaces such as 1xEV-DO that facilitate wireless data need to incorporate support for the varying QoS requirements of voice and data applications. The stringent delay requirements of compressed voice used in wireless telephony can be met by simply giving voice calls priority over data. While network operators would benefit from allocating system resources unused by voice users to data traffic, prioritized voice naturally limits the bandwidth and time slots available to data users. Data throughput, however, can be significantly improved by using scheduling algorithms that exploit multiuser diversity gain.

The main contribution of this paper is a complete analytical characterization of the scheduled rates, delays and packet service times experienced by data users multiplexed with prioritized voice users in a packet-switched airlink. The analytical results in this paper provide useful tools for a network operator to evaluate whether a given mix of high priority voice users and lower priority data users achieves the desired performance objectives for the data users in terms of throughput and delay.

References

1. Bender, P., Black, P., Grob, M., Padovani, R., Sindhusayana, N., Viterbi, A.: CDMA/HDR: A Bandwidth Efficient High Speed Wireless Data Service for Nomadic Users: IEEE Comm. Magazine 70–77 (Jul 2000)
2. TIA/EIA IS-856: CDMA 2000: High Rate Packet Data Air Interface Spec. (2000)
3. Qualcomm 1xEV-DO Quality of Service: White Paper (2003):
http://www.qualcomm.com/cdma/1xEV/media/web_papers/wp-QoS.pdf
4. Knopp, R., Humblet, P.: Information Capacity and Power Control in Single Cell Multiuser Communications: IEEE ICC '95 (Jun 1995)
5. Tse, D. N. C.: Optimal Power Allocation over Parallel Gaussian Channels: Proc., ISIT, Germany (1997)
6. Srinivasan, R., Baras, J.: Opportunistic Scheduling in Wireless Systems and TCP Performance: Technical Report, TR 2002–48, I.S.R. Univ. of MD (Oct 2002)
7. Chan, M. C., Ramjee, R.: TCP/IP Performance over 3G Wireless Links with Rate and Delay Variation: Proc., Mobicom '02 (Sep 2002)
8. Srinivasan, R., Baras, J.: Understanding the Trade-off between Multiuser Diversity Gain and Delay – an Analytical Approach: accepted, IEEE VTC2004-Spring (2004)
9. Rappaport, T.S., Annamalai, A., Buehrer, R. M., Tranter, W. H.: Wireless Communications : Past Events and a Future Perspective: IEEE Comm. Magazine, vol 40, 148–161 (May 2002)
10. Andrews, M., Kumaran, K., Ramanan, K., Stoytar, A.L., Vijayakumar, R., Whiting, P.: CDMA Data QoS Scheduling on the Forward Link with Variable Channel Conditions: Bell Labs Technical Report (April 2000)
11. Shakkottai, S., Stoytar, A. L.: Scheduling for Multiple Flows Sharing Time-varying Channel : The Exponential Rule: American Mathematical Society Translations, Series 2, A Volume in Memory of F. Karpelevich, Vol. 207, (2002)
12. Wu, D., Negi, R.: Effective Capacity – A Wireless Channel Model for Support of Quality of Service: (to appear) IEEE Trans. on Wireless Networking (2003)
13. Liu, X., Chong, K. P., Shroff, N.B.: A Framework for Opportunistic Scheduling in Wireless Networks: Computer Networks, vol. 41, no. 4, 451–474 (2003)
14. 3GPP2 1xEV-DV Evaluation Methodology (2001)
15. Jakes, W. C.: Microwave Mobile Communications: IEEE Press, NJ (1993)

Adaptive Multimedia Streaming for Heterogeneous Networks

Jari Korhonen

Nokia Research Center, Audio-Visual Systems laboratory
P.O. Box 100
33721 Tampere, Finland
jari.ta.korhonen@nokia.com

Abstract. High-quality multimedia streaming applications have strict and often contradicting requirements for network characteristics. Data should be transported in real-time, which restricts usage of retransmissions and buffering. On the other hand, quality should be kept as high as possible. This is why streaming systems should adapt to different network conditions when problems occur. Especially in a heterogeneous IP network infrastructure comprising both wired and wireless components it is often very difficult to achieve optimal trade-off between latency, quality and network overhead, because selection of optimal strategy for end-to-end streaming depends highly on the type of the underlying network. In this paper we address several strategies to optimize streaming applications for different network conditions. In addition, we propose performance evaluation mechanisms to find the bottleneck in the network with minimal modifications to the current mainstream protocol implementations. We also outline an example streaming system utilizing the proposed adaptive technologies to optimize its performance in varying network conditions.

1 Introduction

Latest advances in telecommunications are leading towards convergence between fixed and wireless networking, making wireless access networks virtually an integral part of the traditional Internet. This sets new challenges especially for multimedia streaming applications. In spite of the fast development in wireless networking, available bit rates in cellular systems are still far from those we have in today's high-speed wireline access networks. In addition, the error characteristics of wireless links require different schemes for optimal protocols. In traditional fixed IP networks packet losses are typically caused by congestion. In this case the optimal strategy is to reduce the transport rate when the packet loss rate increases. In contrast, a wireless environment typically suffers from packet losses caused by physical transmission errors. In this case there is no need for aggressive congestion avoidance, and retransmissions or redundant data transport could be used to facilitate recovery from packet losses.

Fig. 1 shows a simplified example of a heterogeneous network infrastructure consisting of subnets and links using different technologies. Large institutions have their own Local Area Networks (LANs) or intranets where servers and workstations are

connected. Home users and small enterprises are attached to the Internet via their Internet service provider's access network using digital subscriber line technologies, such as ADSL. Mobile users can access the net from cellular networks. Wireless LANs (WLANs) are also getting more and more popular, providing ubiquitous Internet access at airports, cafés and universities. Each subnet may use fundamentally different technologies.

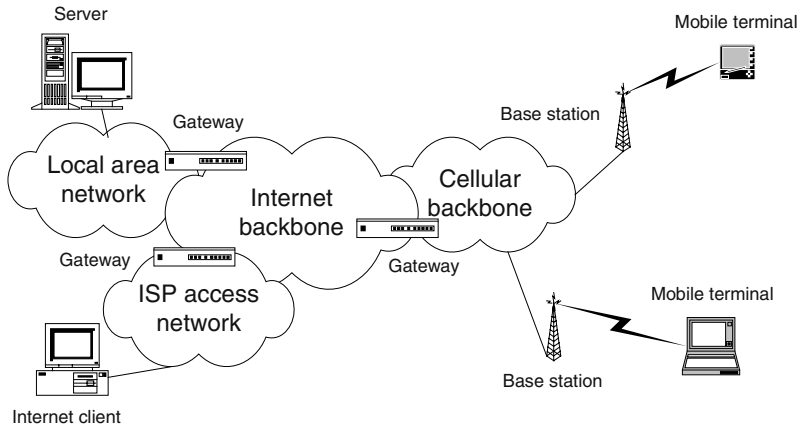


Fig. 1. Modern all-IP network infrastructure merging wired and wireless networks.

Different subnets have different technical characteristics and a streaming system should take this divergence into consideration. In this paper we explain different application and transport layer concepts that can be utilized to find the optimal balance between network resource utilization, quality of the multimedia playback and end-to-end latency. The main focus is on wireless network characteristics and optimizations, including technologies such as packet size optimization, partial retransmissions and Forward Error Correction (FEC). We also discuss how the mainstream IP protocol suite and socket programming interfaces can be used to gain useful information about the circumstances in the underlying physical networks.

2 Fixed vs. Wireless – Data Transport Perspective

The Internet Engineering Task Force (IETF) protocol specifications do not specify in detail the link and the physical layer below the network layer implementing the IP protocol. Some knowledge about the characteristics of the underlying physical network is essential for the design of optimal transport and application layer protocols. In this section we provide an overview of the traditional concepts for optimizing data transport in wired and wireless networks.

2.1 Streaming over Traditional Internet

Congestion control is an essential part of the traditional transport protocol designed for reliable communications over IP networks, namely the Transport Control Protocol (TCP). In practice, packet loss is supposed to indicate congestion in the network and this is why the conventional TCP slows down the transport rate when missing packets are detected, decreasing the total network load [1]. This strategy is obviously valid for applications with no strict requirements for the transport rate, for example file downloading and web browsing.

However, real-time applications like multimedia streaming do not tolerate big changes in transport rate. Therefore streaming applications rely typically on unreliable transport protocols, especially User Datagram Protocol (UDP) and Real-time Transport Protocol (RTP) [2] using normally UDP for actual data transport. This kind of applications keep on sending data at the required rate and whenever a datagram is discarded, it is lost forever. The user may experience packet losses as discontinuations or distortion in the multimedia output, but this is not fatal if the packet loss rate is reasonably low.

Advanced multimedia streaming systems may use application layer congestion control techniques. RTP has been designed to work in parallel with Real-time Transport Control Protocol (RTCP), that conveys statistical information about the quality of the connection between the communicating parties. If congestion is detected, the system can adjust the bit rate to acceptable level compromising between multimedia compression rate and quality.

If the system is streaming live content, the target bit rate can be adjusted by changing the encoder parameters on fly. When prerecorded and encoded multimedia content is handled, the situation is more difficult as the bit rate has typically been defined already during the encoding process. The simplest method is to store different versions of the same content on the server, each representing different bit rate and quality. Then the server can switch to the required bit rate by selecting the most appropriate of the content files.

Another method is to use layered or scalable coding methods. Especially for video coding many layered coding schemes have been proposed in the literature and even adopted by the MPEG-4 standard [3][4]. An encoder using layered coding divides each video frame in a base layer and enhancement layers. Basic quality at low bit rate is available by transmitting base layer data only. If higher quality is required and there are more network resources available, one or more enhancement layers can also be transported to improve the quality. Scalable coding is based on same kind of idea, but it provides finer granularity; some data can be left out from each encoded frame and the frame is still be decodable, only quality is lower [5].

2.2 Streaming over Wireless Channels

In a wireless environment packet losses are often caused by physical errors in the radio channel, appearing as bit errors above the physical layer. Conventionally, both TCP and UDP use checksums to detect bit errors and discard all damaged packets, if

not already discarded by link layer error detection mechanisms. In this case there it is apparently not necessary to adjust the transport rate to reduce the network load. For reliable communications there are a lot of proposals to make TCP more suitable for wireless networking [6][7]. These mechanisms do not aggressively cut down the transport rate immediately when a packet loss occurs but first try to find out whether the packet losses really are caused by congestion.

RTP retransmissions can be useful especially for wireless streaming applications. In a typical multimedia stream packets often have different relative priorities - for example, base layer data has higher priority than enhancement layers. The server can decide to retransmit only the most important packets. Hence the retransmission overhead can be efficiently restricted. Although this is not supported by the conventional usage model of RTP, there is a proposal for extending RTP with selective retransmissions [8].

Simple audio or video codecs tolerate a reasonable number of bit errors by nature. On the other hand, more efficient multimedia compression standards are typically highly vulnerable against bit errors, but different methods can be used to improve the error robustness. Examples of error resilient coding and packetization schemes for Advanced Audio Coding (AAC) are given in [9][10]. As it is often better to have damaged data than no data at all, one possibility is to allow delivery of packets containing bit errors up to the application layer. If a substantial amount of packet losses are caused by bit errors, this approach can decrease packet loss rate significantly.

Traditional UDP specifications provide an option to turn off error checking, but it is not recommended, because it would leave also the UDP header unprotected. UDP Lite proposed in [11] is an elegant, but not yet widely supported solution for this problem. It relies on assumption that only packet headers and a small part of the encoded multimedia content is error sensitive. UDP Lite replaces conventional UDP checksums with partial checksums: only portion of the data in the beginning of the packet payload is protected against errors. If there are bit errors in the unprotected area, the packet is not dropped but the data is conveyed up to the application layer in spite of errors.

2.3 Optimizations for Wireless Networking

A lot of research has been carried out to optimize Medium Access Control (MAC) layer protocols for wireless communications. The most traditional techniques include Forward Error Correction (FEC) and MAC layer retransmissions. Because large packets are more likely to be hit by bit errors, various solutions for adapting MAC layer transport unit (packet) size for wireless channel have been proposed in the literature for optimal trade-off between header overhead and packet loss rate [12][13]. In contrast, application level packet size optimization has not been that intensively studied, because multimedia bitstream formats do not usually allow arbitrary fragmentation of data. For example, if a video or audio frame is split into multiple parts, loss of one part may make all the related parts useless as well.

In practical networks bit errors rarely occur individually, but they are more likely to be clustered as error bursts. This makes weak FEC methods useless. On the other

hand, strong FEC would cause significant redundancy overhead. One possibility to reduce both retransmission and header overhead is to use partial retransmissions. Then each packet is divided in subblocks that are protected by individual checksums. If errors are detected, only damaged subblocks are repeated. This kind of mechanism has been proposed in [14].

3 Optimizing Streaming for Different Conditions

In a practical network infrastructure consisting of both a wireless access network and a wired backbone it is not always a trivial task to select the proper mechanism for end-to-end data streaming. In this section we discuss about different schemes for streaming and guidelines for selection between them in inconsistent circumstances.

3.1 Options for Streaming Adaptation

Very generally speaking, we can separate two clearly different cases where streaming adaptation is needed:

- I) Bottleneck is the capacity of the network: packet losses occur primarily because of congestion in the network devices.
- II) Bottleneck is the physical (wireless) channel: packet losses are caused primarily by bit errors.

Apparently, in case I the only appropriate strategy is to cut the transmission rate and accept weaker quality as penalty. It is also reasonable to use large packets to minimize the header overhead. In case II we have more options. If requirements for latency are not extremely high, selective RTP retransmissions can be considered. If the bitstream format allows packet size to be adjusted smoothly, packet error rate can be reduced by decrementing the average packet size. When delivery of unprotected datagrams is supported, there are even more alternatives as FEC, robust bitstream formats or partial retransmissions could be applied to recover from bit errors instead of packet loss.

If there is more information available of the statistical distribution of bit errors, we can divide case II in subcases to make more precise decisions. Short error bursts can be corrected with FEC or partial retransmissions, i.e. only damaged blocks in each packet are retransmitted. Longer bursts cause more severe damage for packets, but the total number of damaged packets is lower. In this case reasonable strategy is to use smaller packet size to decrease packet loss probability for individual packets, full checksum protection and selective RTP retransmissions.

Sometimes it is possible to extract individual data elements, such as spectral coefficients, from each frame and interleave them among multiple packets. This approach provides high robustness against both packet loss [15] and bit errors [10], not depending on the bit error distribution. However, this scheme does not suit well for highly interactive applications due to the buffering delay needed for interleaving.

3.2 Evaluation of Network Conditions

The selection of appropriate streaming strategy does not only depend on available options provided by the platform, but also on the information fetched about the network conditions. How can a streaming application determine whether the bottleneck of network performance is in wired or wireless part of the network infrastructure?

If the IP protocol stack is accessed via a traditional programming interface, such as Berkeley sockets API for Unix or Winsock API for MS Windows, there are no means to get direct information about the conditions in all the physical networks along the path from the sender to the receiver. To implement a protocol for conveying this kind of information would require radical changes to the established protocol stacks and programming interfaces, because information should be delivered both vertically (cross-layer) and horizontally to cover all subnets.

Anyway, there is still hope. Valuable information about network characteristics can be achieved by analyzing the loss rates for packets with different size. When packet losses are primarily congestion based, there is typically no significant difference between packet loss rates for small and large packets. This is because most routers do not pay attention to the length of the incoming packet, but discard packets automatically when the buffers get full. In contrast, packet loss rate would typically be higher for large than small packets if losses are derived from bit errors.

In a radio channel there are many different independent sources of physical transmission errors, such as interference, multipath propagation and fading. The distribution of arrivals for randomly occurring errors caused by different individual factors can be most easily modeled as a Poisson process, when the error burst occurrences are supposed to follow similar distributions as data packet or phone call arrivals in classical traffic theory. If we assume a radio transmitter to send a continuous stream of bits with constant intervals, the probability p of at least one bit error to occur during transmission of a packet containing l bits can be resolved with (1):

$$p(l) = 1 - \exp(-\lambda l) \quad (1)$$

In (1) $1/\lambda$ is the average number of bits between two errors. However, in this model bit errors are assumed to appear individually. In more precise model we have to take the burstiness of errors into account as well. If we assume that there are homogenous bursts with high error density and the bit error rate is BER , probability to start packet transmission during a bit error burst is approximately equal to BER . We can now see two different cases when a packet is discarded: first, packet transmission starts during an error burst, and second, packet transmission starts without errors, but a new error burst appears during the transmission. Thus we get an equation to evaluate packet loss rate by combining the two cases (2):

$$PLR(l) = BER + (1 - BER) \cdot (1 - \exp(-\lambda l)) \quad (2)$$

There smaller λ indicates higher density of bit error bursts when BER remains the same. The equation (2) can be simplified as (3):

$$PLR(l) = 1 - (1 - BER) \cdot \exp(-\lambda l) \quad (3)$$

When packet loss rates are known for two (or more) different packet lengths, BER and λ can be solved from (3). In this way it is possible to get information about bit error distribution just by sending small and large packets continuously and measuring loss rates for them separately.

In practice, many wireless technologies use link layer retransmissions and the equation (3) cannot be used as such. Typically the number of link layer retransmission attempts is limited and also in this case packet loss rate increases when packet size grows. Anyway, each retransmission attempt causes an extra delay. This is one more reason to use small packets size for delivering time-critical data over a wireless access network.

If the platform supports UDP Lite or delivery of unprotected UDP datagrams, we have even more powerful tools for network analysis. A basic method is to send the most error resilient parts of the multimedia data in unprotected UDP datagrams and compare the packet loss rates of the protected and unprotected packets. If the loss rate is significantly higher for protected than unprotected packets, majority of packet losses are obviously caused by bit errors.

If UDP Lite is supported, it is often reasonable to protect only the protocol headers with the UDP checksum in every packet. Bit errors in the packet payload can then be detected using application layer checksums. This is how the application can know directly if there are bit errors present. It is up to the application how to deal with the erroneous data: more robust parts of the payload could be decoded as usual, but the application can also discard data if the critical parts are hit by errors.

Fig. 2 shows an example of application layer bit error detection using separate Cyclic Redundancy Check (CRC) checksums for different sections of payload data. Protocol headers are protected by UDP Lite partial checksum. There are many advantages in dividing payload in sections with their own checksums. First, error probabilities can be solved separately for sections with different length and equation (1) can be applied to get more exact information about burstiness of bit errors. Second, Unequal Error Protection (UEP) can be easily implemented when payload data is arranged in different sections on the base of the error sensitivity. Third, partial retransmissions can be easily used to replicate damaged sections only.

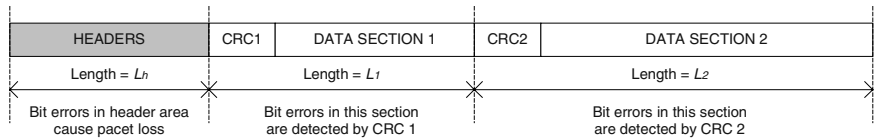


Fig. 2. Application layer bit error detection for unprotected datagrams

4 Putting It All Together – A System Level Perspective

So far we have discussed application and transport layer methods to achieve optimal performance for multimedia streaming in different network conditions as well as

means to reveal the bottleneck in the end-to-end chain of subnets. However, an intelligent streaming system should be able to use these methods together to provide optimal performance in various circumstances adaptively.

4.1 Deciding between Different Options

An intelligent adaptive streaming system uses flexibly the schemes presented in this paper to select always the most appropriate schemes to follow for data transport. Fig. 3 shows a decision tree for selecting the scheme, mainly just summarizing the options explained in the previous section. However, in practice there are very few applications that could use this decision tree example as such. There are a plenty of application dependent factors influencing the decision-making process: what kind of trade-off between quality and network utilization is required, how well does the used bit-stream format suit for packet size adjustment etc.

From the network point of view the world is not black and white either. For example, bit errors and congestion can appear in parallel. In this case congestion should be ranked as the primary problem and retransmissions and FEC should not be used, because traffic jam causes problems also for other users of the network. However, if the codec supports stepless control of transport rate, it might be possible to find an optimal combination of different techniques for every case.

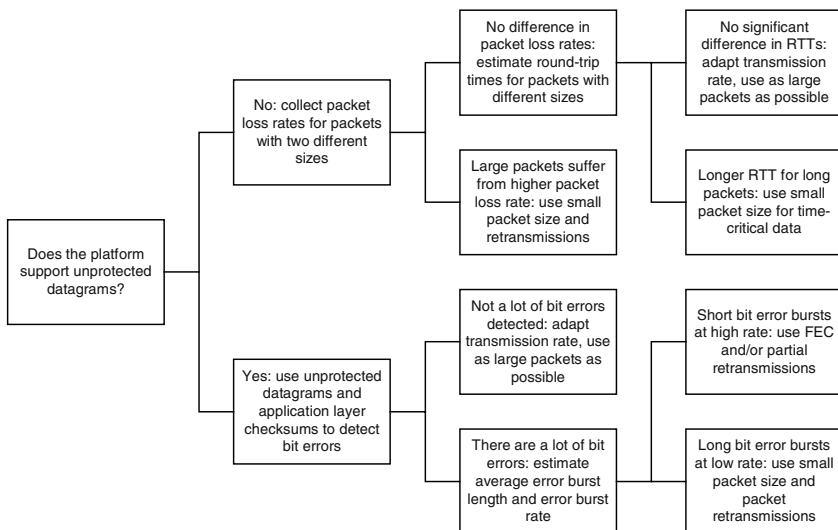


Fig. 3. Decision tree for selecting relevant options for multimedia streaming.

4.2 Design of an Example Streaming System

As discussed above, the requirements for the application, codec type and services provided by the platform limit the possibilities to use the proposed technologies to optimize streaming. Anyway, a generic example system can be outlined. The example configuration consists of a streaming server transporting requested live content to a mobile client. The system does not need to be fully aware of the underlying access technologies; most likely the server is connected to the fixed Internet and the mobile client to the service provider's access network via a wireless LAN, Bluetooth or equivalent. A block diagram of the essential software components for the example streaming server and client are shown in Fig. 4. Here we assume that the platform supports both protected and unprotected delivery of user datagrams.

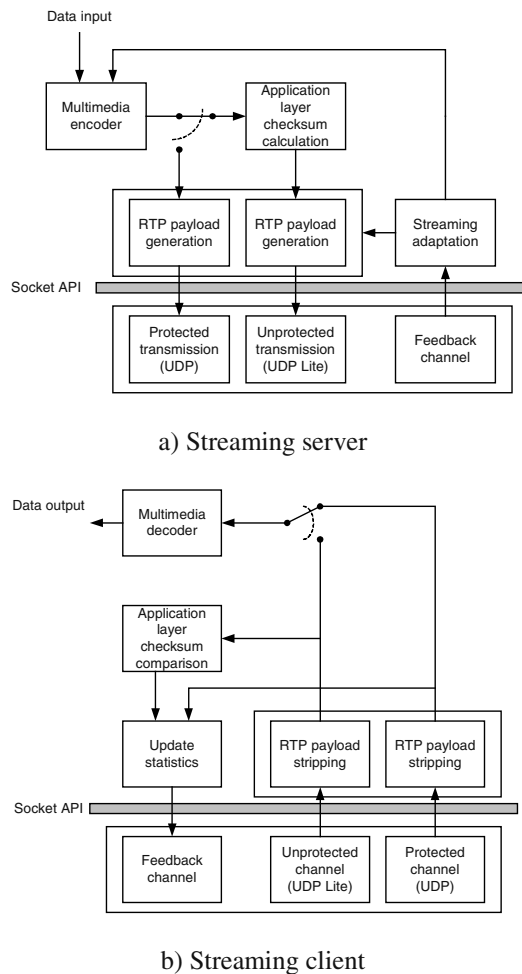


Fig. 4. Block diagram of an example streaming server and client.

In this example multimedia content is encoded while streaming, which allows encoding parameters - essentially the target bit rate - to be changed on-fly. In practice, this is equivalent to rate adaptation with prerecorded content encoded by a scalable or layered compression method.

The server uses both protected and unprotected datagrams for transporting the encoded multimedia data. Most vulnerable and critical parts of the data are protected and parts with higher resilience against bit errors are not. It depends highly on the codec type how this division is made. Bit errors in the unprotected datagrams are detected by application layer checksums.

The streaming client collects statistical data for packet losses, detected bit errors and variance in transport delays (jitter). Traditional RTP provides sufficient means for computing the packet loss rate and jitter. The statistics are conveyed to the server via a feedback channel. The system may use either RTCP (application defined messages) or a proprietary protocol to transport feedback messages.

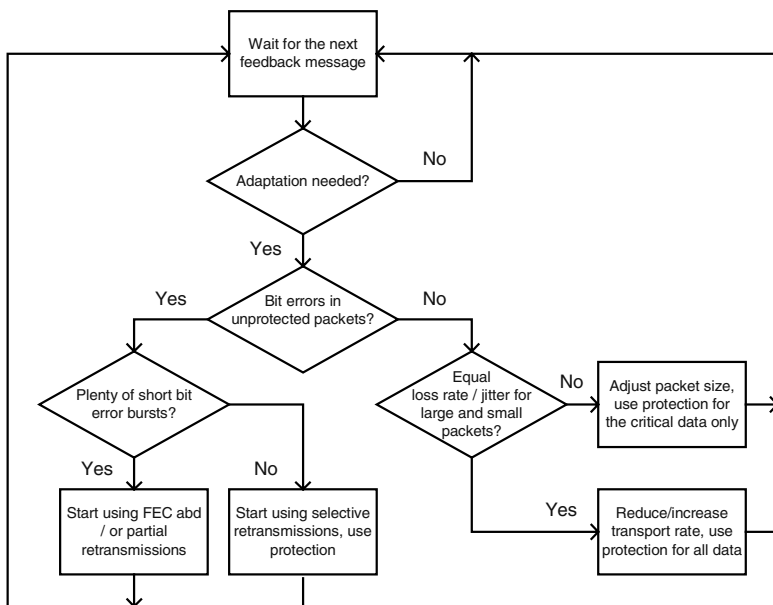


Fig. 5. Flowchart of the decision process for adaptive streaming.

Streaming adaptation is based on the feedback messages. The server makes the decision between different alternatives. If congestion is detected, the encoded bit rate is reduced by changing the encoder parameters and the RTP packet size is maximized, still following the codec-dependent rules for payload packetization. It is also possible that large packets suffer from higher loss rate or jitter than small packets; in this case at least the most critical data should be put in smaller packets to improve the performance. If the problems are diagnosed to come from bit errors, packet size for the most

critical data is reduced or unprotected datagrams with FEC are heavier utilized. When selective or partial RTP retransmissions are allowed, they can be taken in use as well. The example flowchart in Fig. 5 illustrates the decision-making process on the server. It is good to note that even if protection and packet size maximization were recommended, the system should still keep on sending some small and unprotected packets to be able to monitor the network conditions efficiently.

5 Conclusions

In this paper we have addressed problems of multimedia streaming over a heterogeneous IP network infrastructure consisting of both wireless and wired subnetworks. We have shown that distinctively different functionality is required from adaptive transport and application layer mechanisms to find optimal balance between latency, network resource consumption and multimedia quality, depending on whether the bottleneck of the performance is in wired or wireless part of the network. To put it brief, congestion avoidance plays major role if problems are related to the wireline network, whereas packet size optimization and delivery of data in unprotected datagrams are more relevant approaches if a wireless link causes the problems.

We have also explained guidelines for diagnosing the network conditions with the means provided mainly by the traditional protocol architectures and programming interfaces. First, it is possible to transport packets of distinctively different sizes to collect packet loss statistics separately for small and large packets. If large packets suffer from clearly higher loss rate or latency, we may assume that the bottleneck is in the wireless part of the network. Second, if the platform supports delivery of datagrams with bit error detection turned off, we can use application level checksums to find out whether the unprotected packets suffer from bit errors. Finally, we have presented example architecture for a practical streaming system. It uses the methods explained to gain information about network conditions and adapt its functionality to the prevailing circumstances.

References

1. Stevens, W. R., TCP Slow Start, Congestion Avoidance, Fast Retransmission, and Fast Recovery Algorithms. IETF RFC 2001, 1997.
2. Schulzrinne, H., Casner, S., Frederick, R., and Jacobsen, V.: RTP: A Transport Protocol for Real-Time Applications. IETF RFC 1889, 1996.
3. Khansari, M., Zakauddin, A., Chan, W-Y., Dubois, E., and Mermelstein, P.: Approaches to Layered Coding for Dual-Rate Video Transmission. In *proc. of the IEEE Conference on Image Processing (ICIP '94)*, pp. 258–262, 1994.
4. Wollborn, M., Moccagatta, I., and Benzler, U.: Natural Video Coding. In *The MPEG-4 Book*, Pereira, F., and Ebrahimi, T. (ed.), Prentice Hall, Upper Saddle River, New Jersey, pp. 293–381, 2002.

5. Rajendran, R., van der Schaar, M., and Chang, S-F.: FSG+: Optimizing the Joint SNR-Temporal Video Quality in MPEG-4 Fine Grained Scalable Coding. In proc. of the IEEE International Symposium on Circuits and Systems (ISCAS '02), vol. 1, pp. 445–448, 2002.
6. Balakrishnan, H., Padmanabhan, V. N., Srinivasan, S., and Katz, R. H.: A Comparison of Mechanisms for Improving TCP Performance over Wireless Links. *IEEE Transactions on Networking*, vol. 5, no. 6, pp. 756–769, December 1997.
7. Cheng, P. W., and Liew, S.: TCP VenO: TCP Enhancement for Transmission over Wireless Access Networks. *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 2, pp. 216–228, February 2003.
8. Rey, L., Leon, D., Miyazaki, A., Varsa, V., and Hakenberg, R.: RTP Retransmission Payload Format. Internet draft, March 2003.
9. Miao, L., Lu J., and Gu, H.: An improved error resilience scheme for transmission of MPEG-4 audio over EGPRS. *IEEE VTS 54th*, vol. 1, pp. 414–417, 2001.
10. Korhonen, J. and Järvinen, R.: Packetization Scheme for Streaming High-Quality Audio over Wireless Links. To appear in Proc. of Workshop on Multimedia Interactive Protocols and Systems (MIPS '03) 2003, Naples, Italy, November 2003.
11. Larzon, L., Degermark M., and Pink, S.: UDP Lite for Real-Time Multimedia Applications”, in proc. of the IEEE International Conference of Communications (ICC '99), Vancouver, Canada, June 1999.
12. Sarraf, M.: Effect of Slot Size on TDMA Performance in Presence of Per Slot Overhead. Proc. of IEEE Global Communications Conference (GLOBECOM '89), Dallas, Texas, vol.1, pp.604–610, November 1989.
13. Chien, C., Srivastava, M. B., Jain, R., Lettieri, P., Aggarwal, V., and Sternowski, R.: Adaptive Radio for Multimedia Wireless Links. *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 5, pp. 793–813, May 1999.
14. Cheng, H. S., Fairhurst, G., Samaraweera, N.: Efficient Partial Retransmission ARQ Strategy with Error Detection Codes by Feedback Channel. *IEE Proceedings of Communications*, vol. 147, no. 5, pp. 263–268, October 2000.
15. Korhonen, J., and Wang, Y.: Schemes for Error Resilient Streaming of Perceptually Coded Audio. Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2003, Hong Kong, vol. 5, pp. 740–743, April 2003.

A Scalable and Adaptive Key Management Protocol for Group Communication

Yacine Challal, Hatem Bettahar, and Abdelmadjid Bouabdallah

Universite de Technologie de Compiègne
Heudiasyc lab. France
{ychallal,hbettaha,bouabdal}@hds.utc.fr

Abstract. Multicasting is increasingly used as an efficient communication mechanism for group-oriented applications in the Internet. In order to offer secrecy for multicast applications, the traffic encryption key has to be changed whenever a user joins or leaves the system. Such a change has to be communicated to all the current group members. The bandwidth used for such rekeying operation could be high when the group size is large. The proposed solutions to cope with this limitation, commonly called *1 affects n* phenomenon, consist of organizing group members into subgroups that use independent traffic encryption keys. This kind of solutions introduce a new challenge which is the requirement of decrypting and reencrypting multicast messages whenever they pass from one subgroup to another. This is a serious drawback for applications that require real-time communication such as video-conferencing. In order to avoid the systematic decryption / reencryption of messages, we propose in this paper an adaptive solution which structures group members into clusters according to the application requirements in term of synchronization and the membership change behavior in the secure session. Simulation results show that our solution is efficient and more adaptive compared to other schemes.

Keywords: *Security, Multicast, Key Management, Scalability*

1 Introduction

Multicasting is an efficient communication mechanism for group-oriented applications such as video conferencing, interactive group games and video on demand. IP multicast saves bandwidth by sending the source traffic on a multicast tree that spans all the members of the group. Security concerns for IP multicast are more complex than for unicast because of the important number of communicating participants. This raises the problem of group communication confidentiality and thus group key management. Group communication confidentiality requires that only valid users could decrypt the multicast data even if the data is broadcast to the entire network. We assume in what follows that data is encrypted to ensure confidentiality using a symmetric cryptosystem (such as DES [15] or AES [16]). In this case, a symmetric key is used to encrypt data

by the source and to decrypt it by receivers. This key is generally called Traffic Encryption Key (TEK). The confidentiality requirements can be translated into the following key distribution rules [18]:

- Non-group confidentiality : users that were never part of the group should not have access to any key that can decrypt any multicast data sent to the group.
- Forward confidentiality : users which left the group should not have access to any future key. This ensures that a member cannot decrypt data after it leaves the group.
- Backward confidentiality : a new user that joins the session should not have access to any old key. This ensures that a member cannot decrypt data sent before it joins the group.
- Collusion freedom : any set of deleted users should not be able to deduce the current used key.

In order to meet the above requirements, a rekey process should be triggered after each join/leave to/from a secure group. It consists in generating a new TEK and distributing it to the members including the new one in case of a join operation or to the residual members in case of a leave operation. This process ensures that a new member can not decrypt eventually stored multicast data (before its joining) and prevents a leaving member from eavesdropping future multicast data. A critical problem with any rekey technique is scalability : as the rekey process should be triggered after each membership change, the number of TEK update messages may be important in case of frequent join and leave operations. Some solutions propose to organize the secure group into subgroups with different local keys. This reduces the impact of the key updating process, but needs decryption and reencryption operations at the border of subgroups. These operations may decrease the communication quality. We propose to classify current group key management proposals into two approaches:

Approach A : In this approach [11][12][21][20][3][19] all group members share a unique single symmetric key called the Traffic Encryption Key (TEK). This TEK is used by the source to encrypt multicast messages and by the receivers to decrypt them. This approach is mainly used within a centralized architecture where a single key server is responsible for generating and redistributing the new TEK whenever a member joins or leaves the group. Protocols within this approach do not meet scalability requirements since the number of transmitted messages to update TEK is proportional to n , where n is the number of group members [11][12][20] (in the best case to $\log(n)$ [20][21]). This is known as the *1 affects n* phenomenon [13] where a single group membership change (join or leave) results in a rekeying process that disturbs all group members to update TEK. In addition, the use of a single key server leads to a bottleneck problem during TEK distribution and suffers from a single point of failure. Some distributed solutions [4][14] are proposed to share rekeying process among different entities and thereby to cope with scalability, bottlenecks and fault tolerance issues, but they still suffer from the *1 affects n* phenomenon.

Approach B : in order to cope with the approach A drawbacks (*1 affects n*, scalability, bottlenecks), in this approach the multicast group is divided into multiple subgroups[13][7][8][9][17]. Each subgroup shares a local TEK managed by a special entity : the subgroup controller. The set of protocols proposed within this approach (called also : hierarchical key management protocols) are more scalable than centralized protocols. They also attenuate the *1 affects n* problem. In fact, if a member joins or leaves the group, only the concerned subgroup updates its local TEK. However, this improvement is not for free: as subgroups have different TEKs, multicast messages should be decrypted and reencrypted by subgroup controllers whenever they pass from a subgroup to another.

1.1 Motivation

We notice that both approaches suffer from great concerns depending on group dynamicity: *Approach A* suffers from *1 affects n* phenomenon especially if the group is highly dynamic. Whereas, *approach B* suffers from the burdensome decryption / reencryption task, especially if the group is quite static. In this work, we present the Scalable and Adaptive Group Key Management (SAKM) approach that takes advantage of both approaches *A* and *B* by dynamically adapting the key management process with respect to the frequency of membership changes. Within the same secure multicast session, SAKM begins with an *approach A* behavior (a single shared TEK) and dynamically partitions the subgroups into clusters with different local TEKs. The partitioning aims to minimize both decryption / re-encryption and rekeying overheads according to the membership behavior. The partitioning is updated periodically as the membership behavior changes during the multicast session. To illustrate SAKM advantages, let us consider a secure group application (video streaming) that spans a large area (many domains with different countries in different continents). That is, the group members are located in different time beams. This difference in time implies a difference in the membership behavior of the members: the day activity varies from morning to night, a week day activity is more important than a week-end activity, etc...[1]. In this case, it would be interesting to use a protocol that restricts the rekey to the areas with frequent membership changes. Thus, SAKM is very efficient in such situations. Simulation results show that our solution is efficient and more adaptive compared to other schemes.

The remaining of this paper is organized as follows : in section 2 we give an overview of the proposed architecture. Then we present details of the SAKM protocol in section 3 and finally we present the simulation results in section 4.

2 Overview of SAKM Architecture

From the discussion in the previous section, we can conclude that current group key management proposals do not scale well with large and dynamic groups, either because of the *1 affects n* problem or because of the excess of *decryption / reencryption* operations problem. Our approach aims to address these two prob-

lems by taking into consideration the dynamic aspect of the group members. In the SAKM approach, the multicast group is divided into multiple *subgroups* arranged into a tree structure (see figure 1). Each subgroup is managed by a *SAKM Agent* which is responsible for the local key management process. An SAKM Agent have two possible states: *active* or *passive*. An *active* SAKM Agent uses an independent TEK for its subgroup and thus it decrypts and reencrypts received messages before forwarding them to local members. A *passive* SAKM Agent uses the same TEK as its parent subgroup and hence forwards received messages to local members without decryption / reencryption. The whole SAKM Agents' states defines a partition of the subgroups into a set of *clusters*. Each cluster is composed of a set of subgroups that share the same TEK. The cluster's root agent is *active* and all internal agents are *passive*. Messages are decrypted and reencrypted only at the clusters' roots.

Periodically, SAKM Agents exchange dynamism information about their subgroups. Based on these information, each agent estimates two costs: the first cost is the cost of becoming *active* (cost of decryption / reencryption) and the second cost is the cost of becoming *passive* (the *1 affects n* overhead). By comparing the two costs, the SAKM Agent decides whether to become *active* or *passive*. If an agent becomes *passive* it *merges* with its parent cluster and so it uses its parent TEK. If an agent becomes *active* it forms a new *separate* cluster and so uses an independent local TEK.

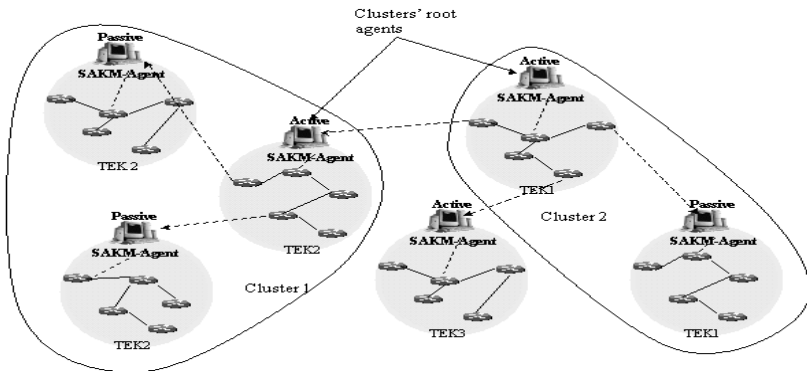


Fig. 1. SAKM Architecture

After each periodic *split* and *merge* process, we obtain a new partition of the group into clusters. This new partition suites better the current membership behavior of the group in terms of both decryption / reencryption cost and *1 affects n* overhead. This way, SAKM approach offers an efficient and adaptive scheme that maintains good performance during the whole secure multicast session. In the following sections, we give detailed description of the SAKM approach.

3 SAKM Protocol

In this section we present our protocol using an elementary system composed of two SAKM subgroups i and j . i is the parent subgroup of j . Each subgroup is managed by an SAKM agent. We suppose that members arrive at a subgroup l with a rate λ_l and stay in the the subgroup an average duration of $\frac{1}{\mu_l}$ seconds. First, we present the criterion which should be used by an agent to decide whether to be *active* or *passive*. The two situations: *active* agent (or *split* subgroups) and *passive* agent (or *merged* subgroups) induce different overheads regarding decryption / reencryption operations and rekeying messages. By comparing the overheads in the two situations, the agent makes the best decision.

Let us consider the overhead induced by decryption/reencryption and rekeying in two cases:

- **case 1:** in the case where two subgroups i and j are merged to use the same keying material. We denote the induced overhead in this case by $(O_{i,j}^{(m)})$;
- **case 2:** in the case where a split operation generates two subgroups i and j and hence each of them uses its own keying material. We denote the induced overhead in this case by $(O_{i,j}^{(s)})$.

An SAKM Agent compares the two quantities $O_{i,j}^{(m)}$ and $O_{i,j}^{(s)}$. If $O_{i,j}^{(m)} > O_{i,j}^{(s)}$ then SAKM agent becomes *active* (i.e. it is more efficient to separate the subgroup from its parent subgroup so that each of them uses its own keying material: a split operation), else it becomes *passive* (i.e. it is more efficient to merge the subgroup with its parent subgroup so that they use the same keying material: a merge operation). In the following, we quantify the overheads $O_{i,j}^{(m)}$ and $O_{i,j}^{(s)}$ in order for an agent to make a decision.

Let $C_{i,j}^{(m)}$ be the average cost of rekeying in the case i and j are merged (case 1). And $C_{i,j}^{(s)}$ the average cost of rekeying in the case i and j are split (case 2). Suppose that the decryption / reencryption overhead depends only on the encryption system (τ), the computation power of the agent that does the operation (P_i) and the rate of messages (r) that characterizes the application traffic. The whole decryption/reencryption overhead is given by $\tau(P_i, r)$.

The overhead induced by *split* subgroups is the sum of both rekeying and decryption / reencryption overheads, thus

$$O_{i,j}^{(s)} = C_{i,j}^{(s)} + \alpha\tau(P_i, r) \quad (1)$$

With (α) the factor characterizing the weight given to a decryption / reencryption operation compared to a rekey message.

The overhead induced by merged subgroups corresponds to the rekeying overhead alone since there is no decryption / reencryption in case of merged subgroups, and thus

$$O_{i,j}^{(m)} = C_{i,j}^{(m)} \quad (2)$$

In case of merged subgroups, each membership change implies a rekey process in both subgroups that share the same TEK. We say that a membership change that occurs in subgroup i has an impact on subgroup j (which is merged with i), and a membership change that occurs in subgroup j has an impact on subgroup i . Therefore, there is a *mutual impact rekeying* in case of merged subgroups because of sharing a same TEK. We denote the mutual impact rekeying cost of subgroup i on subgroup j and vice versa (j on i) by $M_{\{i,j\}}$. Thus, $C_{i,j}^{(m)} = C_{i,j}^{(s)} + M_{\{i,j\}}$ and hence

$$O_{i,j}^{(m)} = C_{i,j}^{(s)} + M_{\{i,j\}} \quad (3)$$

According to 1 and 3, we notice that to compare $O_{i,j}^{(m)}$ and $O_{i,j}^{(s)}$, it is sufficient to compare the two quantities:

$$M_{\{i,j\}} \quad (4)$$

and

$$\alpha\tau(P_i, r) \quad (5)$$

3.1 Overview of SAKM Protocol

We remind that the SAKM architecture is made up of an hierarchy of multicast subgroups. The whole hierarchy forms an SAKM multicast group. The keying material inside a subgroup is managed by an SAKM agent. Two adjacent subgroups may merge to use the same parent's keying material if the mutual impact rekeying cost is less than the decryption / reencryption cost at the child SAKM agent. In the contrary case, the two adjacent subgroups are split and each of them uses its own keying material. The SAKM aim is partitioning the hierarchy into clusters so that both decryption / reencryption and rekeying overheads are minimized. Each cluster is a set of SAKM subgroups using the same keying material. It is well known that finding a partition of a hypergraph (the SAKM tree) with the minimal cost is an NP-Complete problem [10] and because of that, SAKM uses a heuristic to approach the optimal solution. SAKM proceeds then as follows: periodically, each SAKM agent x calculates new estimations of the two parameters λ_x (the mean arrival rate of members at the agent's subgroup) and μ_x (where $\frac{1}{\mu_x}$ is the mean membership duration of the members of the agent's subgroup). x sends these parameters (λ_x, μ_x) to its child SAKM agents (y). Each child SAKM agent (y) compares then the two costs: $\alpha\tau(P_y, r)$ and $M_{\{x,y\}}$. If $\alpha\tau(P_y, r) \leq M_{\{x,y\}}$, then y becomes *active*. If $\alpha\tau(P_y, r) > M_{\{x,y\}}$, then y becomes *passive*.

Each SAKM agent y holds two Traffic Encryption Keys (TEKs): TEK_y used in its own subgroup and TEK_x used in its parent subgroup (x). Note that if y is *passive* then $TEK_y = TEK_x$. If y is *active* then, it decrypts received messages using TEK_x and reencrypts them toward its own subgroup using TEK_y .

Five types of messages are involved in the protocol:

- **NEW_TEK_RQ**: this type of message is sent by a *passive* agent when a membership change occurs in its subgroup. This message specifies the

membership change type (join or leave) and it is sent to the cluster's root agent which is responsible for delivering TEKs for the cluster.

- **IM_ACTIVE**: this type of message is sent by an agent when it becomes *active*. The message is sent to the cluster's root agent. Upon receiving the message, this later distributes a new TEK to be used by the remaining subgroups in the cluster.
- **NEW_TEK**: this message type is used by an *active* agent to distribute a new TEK to its cluster. It specifies the new TEK and the agent's identity which is necessary for the *passive* agents to request new TEKs whenever membership changes occur in their subgroups. It specifies also the membership change type which caused delivering the new TEK, because the distribution scheme of the new TEK depends on the membership change type (join or leave).
- **JOIN_LEAVE**: these messages are sent by the members to join or leave the virtual multicast group. An exclusion of a member by an agent (because the allowed duration expires) is considered as receiving a LEAVE message. This type of message specifies the membership change type along with the required member authentication information.
- **NEW_DYN_INF**: this message is sent by SAKM agents to refresh their dynamicity parameters estimation. Each agent sends this message periodically to its child agents. It specifies the new estimations of the two parameters λ and μ . Upon receiving this message, each child agent decides whether to become *passive* or *active*.

3.2 Merge / Split Protocol

In this phase of SAKM, the hierarchy is partitioned into clusters that use the same keying material. Periodically (let say after each θ units of time), each SAKM agent x sends the new estimations of λ_x and μ_x to its children (let us designate by y any child of x). x signs the message with its private key k_x^{-1} to ensure the authenticity and the integrity of the sent parameters. Upon receiving λ_x and μ_x (the parent's parameters), a child SAKM agent y compares $\alpha\tau(P_y, r)$ to $M_{\{x,y\}}$ and takes the decision to become *active* or *passive* according to the results of that comparison. Figure 2 summarizes this phase.

When y becomes *active*, it generates a new TEK_y and multicasts it to the members of its subgroup (line 1). Note that as y 's child SAKM agents are members in y 's subgroup, they receive the new TEK_y . When a *passive* child SAKM agent receives the new TEK_y , it forwards this key to its subgroup and the process continues until all the members in the cluster that uses y 's TEK receive the new TEK_y . y informs then the cluster's root agent (t) about the decision to become *active* (line 2) using the "IM_ACTIVE" message. Upon receiving the message, t distributes a new TEK_t to the remaining subgroups in the cluster. This update of TEK_t is compulsory to ensure backward and forward secrecy.

In the case where y becomes *passive*, it multicasts its parent's TEK TEK_x to the members of its subgroup (line 4) and changes its state to *passive*. This TEK is forwarded by y 's children to all the members in the y 's old cluster and informs SAKM agents of the cluster about the new cluster's root.

Let x and y be SAKM agents with x parent of y .

Agent x

$x \rightarrow y :$

$\langle NEW_DYN_INF, \lambda_x, \mu_x, timeStamp \rangle_{k_x^{-1}}$ /* periodically */

Agent y

If (state=PASSIVE and $\alpha\tau(P_y, r) \leq M_{\{x,y\}}$) **then**

1. $y \rightarrow SUBG_ADD :$

$\langle NEW_TEK, \{newTEK\}_{oldTEK}, JOIN, myID, timeStamp \rangle_{k_y^{-1}};$

2. $y \rightarrow CLUSTER_ROOT_ADD :$

$\langle IM_ACTIVE, timeStamp \rangle_{k_y^{-1}};$

3. state:=ACTIVE;

end if

if (state=ACTIVE and $\alpha\tau(P_y, r) > M_{\{x,y\}}$) **then**

4. $y \rightarrow SUBG_ADD :$

$\langle NEW_TEK, \{parentTEK\}_{oldTEK}, JOIN, clusterRoot, timeStamp \rangle_{k_y^{-1}};$

5. state:=PASSIVE;

end if

Fig. 2. Split / merge protocol

3.3 Membership Change Protocol

The main idea in this phase is to restrict rekeying to the cluster where occurs the membership change (join or leave). This minimizes the *1 affects n* phenomenon as only the members of the cluster are concerned by the rekey. As the members of a cluster use the cluster's root TEK, when a membership change occurs in a subgroup, the join / leave information is sent to the cluster's root which is responsible for generating and distributing a new TEK for the valid members in the cluster. When a membership change occurs in a subgroup, the SAKM agent responsible for that subgroup reacts as follows:

If the agent is *active* (which means that it is a cluster's root), it generates and distributes a new TEK to its subgroup and hence to its cluster (the distribution is forwarded by its child SAKM agents). If the agent is *passive* (an internal agent in a cluster), it sends a request to the cluster's root asking for a new TEK for the cluster ("NEW-TEK.RQ"). Then the cluster's root agent generates and distributes a new TEK for the cluster. All the agents of the cluster distribute the new TEK to their subgroup members according to their rekey strategy and depending on the membership change type (join or leave).

4 Simulation Results

In this section, we provide an overview of our simulation model and some of the results we obtained by comparing SAKM with Iolus and a centralized solution.

As an approach *B* representative protocol, we selected Iolus [13]. In approach *B*, the multicast group is divided into multiple subgroups, in a static manner, with independent TEKs and thus it suffers from the high number of decryption / reencryption operations. We selected the centralized solution as a representative protocol of approach *A*. In this approach, members share a same TEK and thus suffer from the *1 affects n* phenomenon. We study the *1 affects n* behavior of each simulated protocol and the number of decryption / reencryption operations required for the communication.

4.1 Simulation Model

In our simulation, we use a virtual SAKM multicast group made up of five multicast subgroups. We suppose that the group is composed of 100 dynamic members in the average.

To generate real multicast sessions, we used the models presented by Almeroth et al. in [1][2]. These models suggest that the arrival of members follows a Poisson process and the membership duration follows an Exponential distribution. These models are deduced from real multicast sessions observed on the Mbone. In our simulations, we consider a session of 3 hours, an inter-arrival between members of 20 seconds and an average membership duration of 30 minutes. We suppose that the distribution of arrival members at the different subgroups is not uniform and changes over time.

4.2 Split / Merge Criteria

If we consider a flow with a rate r of 16kbps of encrypted data, and we use DES to assure its secrecy, then per each r Mbytes of received data, we would have an overhead of $2.r.DES_t$ ($r.DES_t$ for decryption and $r.DES_t$ for reencryption). In our simulations, $DES_t = 78ms$ to encrypt 1Mbytes of data (using a Celeron 850 MHz processor [6]). Thus the decryption / reencryption overhead becomes

$$\alpha\tau(P_i, r) = 2.\alpha.r.DES_t$$

We will show that α plays a key role in controlling the behavior of SAKM regarding synchronization requirements of the application.

Consider two adjacent subgroups of an SAKM hierarchy i and j with i parent of j . Suppose that both SAKM agents of subgroups i and j use the hierarchical key graph scheme [21][20] for rekeying and that the graph is a full binary tree. Thus the mutual impact rekeying cost would be

$$M_{\{i,j\}} = 3(\lambda_i + \lambda_j)$$

(see [5] for an explanation of how to calculate $M_{\{i,j\}}$).

Hence, for j to take a decision about its state, it has to compare between the two costs $2.\alpha.r.DES_t$ and $M_{\{i,j\}} = 3(\lambda_i + \lambda_j)$ after each θ units of time (15mins in our simulation).

4.3 Simulation Results and Discussion

We consider an application that does not need real-time data transmission such as replicating distributed data bases, software updating or broadcasting stock quotes. We recall that SAKM aims to minimize decryption / reencryption overhead as well as the *1 affects n* phenomenon by considering the members' dynamism.

Figure 3 shows the results obtained with such an application: (a) measures the number of decryption / reencryption operations which corresponds to the number of clusters in SAKM and to the number of subgroups in Iolus (5 in our case). (b) measures the number of affected members: with the centralized solution all the members are affected, with Iolus only the members of a subgroup are affected, that is why the results of Iolus are much better, SAKM makes a tradeoff between decryption / reencryption overhead and *1 affects n*.

As this type of applications tolerate latencies, we give a small value to the weight α (4 in our case), so that SAKM creates as much clusters it needs to attenuate *1 affects n* and to minimize decryption / reencryption overhead compared to Iolus which makes it systematically at each Iolus agent.

At a first sight, we remark that even if SAKM makes only three decryption / reencryption operations in the average (see figure 3 (a)), it maintains as good performances as those of Iolus (see figure 3 (b)).

SAKM starts with a centralized behavior (a single cluster) ($0 < t < 700s$) and as the group size grows, the group dynamism increases and thus SAKM creates new clusters following the split / merge process; at $t = 700s$ SAKM creates only 3 clusters (see figure 3 (a)) and reaches with that Iolus performances regarding *1 affects n* attenuation (see figure 3 (b)), which means that SAKM saves decryption / reencryption overhead as well as rekeying messages overhead, which is not the case with Iolus.

Whenever the group dynamism reaches a certain degree, SAKM creates a new cluster in order to attenuate the *1 affects n* overhead: (see for example: figure 3 (a) and figure 3 (b) at $3800s < t < 5500s$ and at $6500s < t < 9000s$).

Whenever decryption / reencryption cost ($2.\alpha.r.DES_t$) exceeds the mutual impact rekeying cost, SAKM merges as much clusters to reach a better whole partition cost (see figure 3 (a) and (b) at $t = 5500s$ and $t = 9000s$).

In this way, SAKM assures a tradeoff between decryption / reencryption cost and rekeying cost in order to keep good performances of the whole SAKM architecture.

If we consider an application that requires high synchronization between the source and receivers (such as video streaming), Iolus do not fit well since decryption / reencryption operations would introduce latencies between the source and receivers. However, with SAKM we can adapt the scheme by setting the factor α to a value that prevents SAKM from creating a lot of clusters except for situations where the membership changes in a sharply way and it would be better to limit the rekey to the subgroup(s) where the membership changes. Such a situation is very rare and do not affect performances of the application that

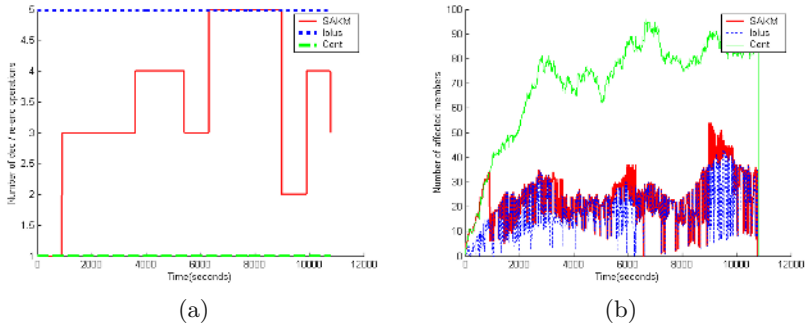


Fig. 3. Simulation results for an application with no synchronization requirement

tolerates some latencies (it is not a serious problem when we receive a slightly slow sequence for a while when seeing a movie on the Internet).

Finally, if we consider an application that requires real-time transmission (such as videoconferencing), where it is out of question to do decryption / reencryption of the data which should reach receivers in real-time, Iolus will not support this kind of requirement. With SAKM we can adapt the scheme by setting the factor α to infinite theoretically to prevent SAKM from creating clusters. And hence SAKM becomes typically a centralized solution without intermediaries.

The simulation results of these two types of applications can be found in [5].

5 Conclusion

Security mechanisms are an urgent requirement for multicasting in order to ensure a safe and large deployment for confidential group communications. In real multicast sessions, members can join and leave the group dynamically during the whole session. This dynamicity affects considerably the performances of the key management protocol. In this paper, we considered a special class of group key management which subdivides the multicast group into subgroups with independent traffic encryption keys so that the *1 affects n* phenomenon and hence the rekeying overhead are minimized. In this kind of architectures, multicast messages should be decrypted and reencrypted at the boundaries of subgroups. The decryption / reencryption operations create a new overhead which could be disastrous for some kind of applications that require a real-time or highly synchronized data transmission. We showed that it is possible to make a tradeoff between the two overheads (decryption / reencryption and rekeying overheads) by making an adaptive clustering of subgroups. As partitioning the subgroups' hierarchy with the minimum overhead is an NP-Complete problem, we proposed a heuristic to solve the problem and the simulation results show that it is a good approach compared to two other approaches from the literature.

References

1. K. Almeroth and M. Ammar. Collecting and modelling the join/leave behaviour of multicast group members in the Mbone. *Symposium on High Performance Distributed Computing*, 1996.
2. K. Almeroth and M. Ammar. Multicast group behaviour in the internet's multicast backbone (Mbone). *IEEE communications Magazine*, 1997.
3. D. Balenson, D. McGrew, and A. Sherman. *Key Management for Large Dynamic Groups : One-Way Function Trees and Amortized Initialization*. draft-balensongroupleykeymgmt-oft-00.txt, February 1999. Internet-Draft.
4. Ghassen Chaddoud, Isabelle Chrisment, and Andre Sha.. Dynamic Group Communication Security. *6th IEEE Symposium on computers and communication*, 2001.
5. Y. Challal, H. Bettahar, and A. Bouabdallah. SAKM: *Analytic Model and Theoretical Constructions*, April 2003. Technical Report.
6. Wei Dai. *Comparison of popular cryptographic algorithms*. <http://www.eskimo.com/~weidai/benchmarks.html>, 2000.
7. Lakshminath R. Dondeti, Sarit Mukherjee, and Ashok Samal. Comparison of Hierarchical Key Distribution Schemes. *IEEE Globcom Global Internet Symposium*, 1999.
8. Lakshminath R. Dondeti, Sarit Mukherjee, and Ashok Samal. *Survey and Comparison of Secure Group Communication Protocols*, 1999. Technical Report.
9. Lakshminath R. Dondeti, Sarit Mukherjee, and Ashok Samal. Scalable secure one-to-many group communication using dual encryption. *Computer Communications*, 2000.
10. M. Gondran and M. Minoux. *Graphs and Algorithms*. Wiley-interscience series in discrete mathematics edition, 1990.
11. H. Harney and C. Muckenhirn. *Group Key Management Protocol (GKMP) Architecture*, July 1997. RFC 2093.
12. H. Harney and C. Muckenhirn. *Group Key Management Protocol (GKMP) Specification*, July 1997. RFC 2094.
13. Suvo Mittra. Iolus : A Framework for Scalable Secure Multicasting. *ACM SIGCOMM*, 1997.
14. Rolf Oppliger and Andres Albanese. Distributed registration and key distribution (DIRK). *Proceedings of the 12th International Conference on Information Security IFIP SEC'96*, 1996.
15. Federal Information Processing Standards Publication. *Data Encryption Standard (DES)*, December 1993. FIPS PUB 46.
16. Federal Information Processing Standards Publication. *Advanced Encryption Standard (AES)*, November 2001. FIPS PUB 197.
17. Clay Shiels and J.J. Garcia-Luna-Aceves. KHIP-A scalable protocol for secure multicast routing. *ACM SIGCOMM*, 1999.
18. Jack Snoeyink, Subhash Suri, and George Varghese. A Lower Bound for Multicast Key Distribution. *IEEE INFOCOM'01*, 2001.
19. M. Waldvogel, G. Caronni, D. Sun, N. Weiler, , and B. Plattner. The VersaKey Framework : Versatile Group Key Management. *IEEE Journal on Selected Areas in Communications (Special Issues on Middleware)*, 17(8):1614-1631, August 1999.
20. D. Wallner, E. Harder, and R. Agee. *Key Management for Multicast: Issues and Architecture*. National Security Agency, June 1999. RFC 2627.
21. Chung Kei Wong, Mohamed Gouda, and Simon S. Lam. Secure Group Communications Using Key Graphs. *ACM SIGCOMM*, 1998.

Smooth Fast Broadcasting (SFB) for Compressed Videos

Hsiang-Fu Yu^{1, 2}, Hung-Chang Yang¹, Yi-Ming Chen³,
Li-Ming Tseng¹, and Chen-Yi Kuo¹

¹ Dep. Of Computer Science and Information Engineering, National Central University,
Jung-Li, Taoyuan, Taiwan, 320
{yu, cyht, kuo jy}@dslab.csie.ncu.edu.tw

² Computer Center, National Central University,
Jung-Li, Taoyuan, Taiwan, 320
center3@cc.ncu.edu.tw

³ Dep. Of Information Management, National Central University,
Jung-Li, Taoyuan, Taiwan, 320
cym@im.mgt.ncu.edu.tw

Abstract. One way to broadcast a popular video is to partition the video into segments, which are broadcasted on several streams periodically. The approach lets multiple users share streams; thus, the stress on the scarce bandwidth can be alleviated without sacrificing viewers' waiting time. One representative approach is the Fast Broadcasting (FB) scheme. The scheme does not obtain shortest waiting time but it can offer a more reliable video transmission on wireless networks. However, the scheme mainly supports transmission of CBR-encoded videos. In this paper, we propose a FB-based scheme for VBR-encoded videos. The scheme can smooth required bandwidth. From the simulation results, the SFB scheme has smaller required bandwidth, buffers, and disk transfer rate than the FB scheme. For a video, the maximum difference of its required bandwidth is less or equal to $\max(B_{\max}^i - B_{\min}^i)$, where B_{\max}^i and B_{\min}^i represent the maximum and minimum required bandwidth on stream i .

1 Introduction

With the advancement of broadband networking technology and the growth of processor speed and disk capacity, video-on-demand (VOD) services have become possible [11]. A VOD system is typically implemented by a client-server architecture, and may easily run out of bandwidth because the growth in bandwidth can never keep up with the growth in the number of clients. To alleviate the stress on the bandwidth and I/O demands, many alternatives have been proposed by sacrificing some VCR functions, or known as near-VOD services. One way is to broadcast popular videos. According to [2], 80% of demands are on a few (10 or 20) very popular videos. Because the server's broadcasting activity is independent of the arrivals of requests, the approach is appropriate to popular or hot videos that may interest many viewers at a certain period of time. One way to broadcast a popular video is to partition the video into segments, which are broadcasted on several streams periodically. The schemes [1], [4], [5], [6], [7], [8], [9], [13], [16], [17], [19] share a similar arrangement. A video

server divides a video into segments that are simultaneously broadcasted on different data streams. One of these streams transmits the first segment in real time. The other streams transmit the remaining segments according to a schedule predefined by the scheme. When clients want to watch a video, they wait first for the beginning of the first segment on the first stream. Thus, their maximum waiting time equals the length of the first segment. While the clients start watching the video, their set-top boxes (STB) or computers start downloading enough data from the other streams so they will be able to play the segments of the video in turn.

The simplest broadcasting scheme is the staggered broadcasting [1]. The pyramid broadcasting [18] partitions a video into increasing size of segments and transmits them on multiple streams of the same bandwidth. It requires less bandwidth than the staggered broadcasting under the same maximum waiting time. The fast broadcasting (FB) [4] divides a video into a geometrical series. In comparison with the staggered broadcasting and the pyramid broadcasting, the FB scheme obtains shorter waiting time. Additionally, the scheme can offer a more reliable video transmission on wireless networks [3]. Based on the pagoda broadcasting scheme, the new pagoda broadcasting (NPB) scheme [13] partitions a video into fixed-size segments and maps them into data streams of equal bandwidth at the proper decreasing frequencies. Accordingly, the NPB scheme obtains shorter waiting time than the FB scheme. The recursive frequency splitting (RFS) scheme [16] further improves the NPB scheme on waiting time by using a more complex segment-to-stream mapping. The harmonic broadcasting (HB) scheme [6] first divides a video into several segments equally, and further divides the segments into sub-segments according to the harmonic series. Yang, Juhn, and Tseng [20] proved that the HB scheme requires the minimum bandwidth under the same waiting time. An implementation of the FB scheme on IP networks was reported in [21].

The above schemes assume that videos are encoded in constant-bit-rate (CBR). Accordingly, they cannot support variable-bit-rate (VBR) videos well. Some schemes were proposed to address this problem. The variable bandwidth harmonic broadcasting (VBHB) [14] first divides a VBR video into fixed size segments. The first and second segments are broadcasted at the transmission rate guaranteeing on time delivery of all frames. All other segments are divided into equal-size sub-segments, which are distributed in the way of the cautious harmonic broadcasting (CHB) scheme [12]. The periodic broadcasting with VBR-encoded video (VBR-B) [15] integrates the pyramid broadcasting scheme with the techniques of the GoP smoothing, server buffering, and client prefetching to transmit VBR videos. Based on the VBR-B, the trace adaptive fragmentation (TAF) scheme [10] takes the trace of each video into account to predict the bandwidth requirements, and then uses complex techniques to smooth the bandwidth consumption.

In this paper, we propose the smooth fast broadcasting (SFB) scheme for VBR-encoded videos. The scheme can also reduce the variance of required bandwidth. It is systematic and simple in concept. A VBR video is divided into multiple equal-length segments by time. To smooth the bandwidth consumption, we first require the video server to transmit each segment at constant bit rate. Thus, the bandwidth requirements for each segment will be constant during its distribution. We then re-arrange the order of the segments on each stream to smooth the total bandwidth requirements. From the

simulation results, the SFB scheme has smaller required bandwidth, buffers, and disk transfer rate than the FB scheme.

The rest of this paper is organized as follows. In Section 2, we present the SFB scheme for VBR videos. Some analysis and simulation results are presented in Section 3. We make brief conclusions in Section 4.

2 Smooth Fast Broadcasting for VBR-Encoded Videos

To smooth the variance of bandwidth requirements for VBR videos, we propose the smooth fast broadcasting (SFB) scheme. The SFB scheme and the FB scheme differ in two areas.

- Asynchronous download and playout. The data consumption rate of a VBR video varies with time so the rate is probably larger than its data transfer rate. In the FB scheme, a client receives and plays a segment concurrently; thus, the video playout may be blocked when the consumption rate is larger than the transfer rate. To ensure the continuous playout, the SFB scheme requires a client to buffer a segment completely before playing it. This restriction causes the SFB scheme having larger average waiting time than the FB scheme; however, the two schemes have the same maximum waiting time.
- Smooth bandwidth requirements. The SFB scheme transmits each segment at constant bit rate. Thus, for each segment, the variance of the required bandwidth is zero. The scheme further rearranges the order of segments on each stream to reduce the variance of total required bandwidth.

On the server side, the SFB scheme involves the following steps.

1. A video, whose length is L , is equally divided into N segments by length, and the number of required streams is

$$K = \lceil \log_2 (N + 1) \rceil \quad (1)$$

2. Apply the segment arrangement algorithm to arrange the order of segments on each stream. Suppose S_i is i th segments, and s_i is its size. Figure 1 illustrates the algorithm by a VBR video, which is divided into 15 equal-length-but-unequal-size segments. Suppose the size of the segments is 3, 4, 9, 5, 3, 2, 7, 5, 4, 9, 2, 5, 7, 5, and 3. In the figure, the rectangles represent the segments of the video, and the area reflects the size of a segment. As the FB scheme, the segments are classified into four groups that are broadcasted on four streams. On stream 1, the bandwidth requirements are constant because the stream always transmits the first segment. On the first two streams, the bandwidth requirements (*i.e.* 7 and 12) vary cyclically. The video server cannot smooth the bandwidth requirements, regardless of the arrangement of segments 2 and 3. In order to smooth the bandwidth requirements for the first three streams, we broadcast the biggest segment on stream 3 together with the segments that totally consume minimum bandwidth on streams 1 and 2. We next group the second biggest segments on stream 3 and the segments that totally consume second minimum bandwidth

on streams 1 and 2. We repeat the arrangement until all of the segments on stream 3 are settled. We also use the same way to arrange the segments on stream 4, and gain the final broadcasting map, as indicated in Fig. 1. The pseudo code of the algorithm details in Fig. 2. The segment arrangement obtained by the SFB scheme stores in an array $V[]$. The segments numbered from $V[2^{i-1}]$ to $V[2^i - 1]$ are broadcasted on stream i sequentially, where $1 \leq i \leq K$.

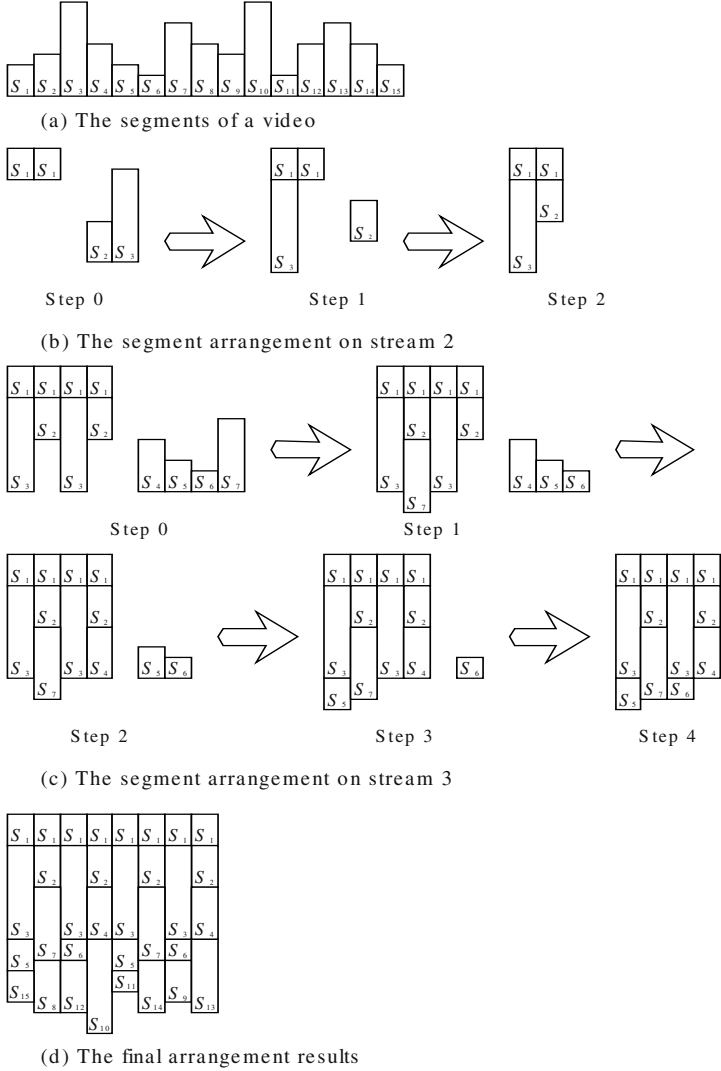


Fig. 1. An example for the segment arrangement by the SFB scheme

3. According to the broadcasting map, the video server transmits the segments periodically.

At the client end, suppose there is plenty of disk space to buffer portions of the playing video. For watching a video, the following steps are involved:

1. Download all of the segments concurrently during each time slot.
2. To ensure a segment was buffered completely before its use, we delay the playout a period time. If the client begins to download the video segments at T_0 , the video can be played in the order of $S_1 \bullet S_2 \bullet \dots \bullet S_N$ at

$$T_0 + \frac{L}{N}.$$

3. Stop loading data from networks when we have received all of the segments.

```

# K is the number of streams.
# S[1..2^K-1] stores the size of the segments. If a segment is blank, its size equals 0.
# B[1..2^K-1] stores the bandwidth consumption of the time slots.
# V[1..2^K-1] stores the new arrangement of video segments
B[1..2^K-1] = 0
B[1] = S[1]
V[1] = 1
If K = 1 then return
For i = 2 to K do
  copy(B[], 1, 2^{i-2}, 2^{i-2} + 1)
  # copy B[1..2^{i-2}] to B[2^{i-2} + 1..2^{i-1}], for example, suppose B[1..4] = {8,3,0,0}.
  # After copy(B[], 1, 2, 3), B[1..4] = {8,3,8,3}
  sort_value_but_return_key(B[], 1, 2^{i-1}, C[])
  # sort B[1..2^{i-1}] and save the keys to C[1..2^{i-1}], for example, B[1..4] = {8,3,8,3}
  # then C[1..4] = {2,4,1,3}
  sort_value_but_return_key(S[], 2^{i-1}, 2^i-1, T[])
  # sort S[2^{i-1}..2^i-1] and save the keys to T[], for example, suppose S[4..7] = {8,5,4,9}.
  # Then, T[1..4] = {6,5,4,9}
  For j = 1 to 2^{i-1} do
    B[C[j]] = B[C[j]] + S[T[2^{i-1} + 1 - j]]
    V[2^{i-1} + C[j] - 1] = T[2^{i-1} + 1 - j]
  End for
End for

```

Fig. 2. The pseudo code of the SFB scheme

3 Analysis and Comparison

3.1 Bound Analysis

We first derive the maximum difference of total bandwidth required during video distribution by the SFB scheme.

Lemma 1: Suppose there are two sorted series $X = \{x_1, x_2, \dots, x_n\}$ and $Y = \{y_1, y_2, \dots, y_n\}$, where $0 \leq x_1 \leq x_2 \leq \dots \leq x_n$ and $0 \leq y_1 \leq y_2 \leq \dots \leq y_n$. Suppose

there is a series $Z = \{z_1, z_2, \dots, z_n\}$, where $z_i = x_i + y_{n-i+1}$. Then, $z_{\max} - z_{\min}$ is less or equal to $\max(x_n - x_1, y_n - y_1)$.

Proof: For any two elements z_j and z_k in Z , from the assumption of Lemma 1, $z_j = x_j + y_{n-j+1}$ and $z_k = x_k + y_{n-k+1}$. $z_j - z_k = (x_j - x_k) + (y_{n-j+1} - y_{n-k+1})$.

Case 1 ($j \geq k$):

$$0 \leq x_1 \leq x_2 \leq \dots \leq x_n \Rightarrow x_n - x_1 \geq x_j - x_k \geq 0 \quad (2)$$

$$j \geq k \Rightarrow -k \geq -j \Rightarrow n - k + 1 \geq n - j + 1$$

$$0 \leq y_1 \leq y_2 \leq \dots \leq y_n \Rightarrow y_{n-k+1} \geq y_{n-j+1} \Rightarrow 0 \geq y_{n-j+1} - y_{n-k+1} \geq y_1 - y_n \quad (3)$$

From equations (2) and (3),

$$\begin{aligned} x_n - x_1 &\geq x_j - x_k + y_{n-j+1} - y_{n-k+1} \geq y_1 - y_n \Rightarrow x_n - x_1 \geq z_j - z_k \geq y_1 - y_n \\ &\Rightarrow \max(x_n - x_1, y_n - y_1) \geq |z_j - z_k| \end{aligned}$$

Case 2 ($k > j$):

$$0 \leq x_1 \leq x_2 \leq \dots \leq x_n \Rightarrow x_n - x_1 \geq x_k - x_j \geq 0 \quad (4)$$

$$k > j \Rightarrow -j \geq -k \Rightarrow n - j + 1 \geq n - k + 1$$

$$0 \leq y_1 \leq y_2 \leq \dots \leq y_n \Rightarrow y_{n-j+1} \geq y_{n-k+1} \Rightarrow 0 \geq y_{n-k+1} - y_{n-j+1} \geq y_1 - y_n \quad (5)$$

From equations (4) and (5),

$$\begin{aligned} x_n - x_1 &\geq x_k - x_j + y_{n-k+1} - y_{n-j+1} \geq y_1 - y_n \Rightarrow x_n - x_1 \geq z_k - z_j \geq y_1 - y_n \\ &\Rightarrow \max(x_n - x_1, y_n - y_1) \geq |z_k - z_j| \end{aligned}$$

From the results, we prove the lemma. \square

Theorem 1: For a video distributed by the SFB scheme, the maximum difference of its required bandwidth is less or equal to $\max(B_{\max}^i - B_{\min}^i)$, where B_{\max}^i and B_{\min}^i represent the maximum and the minimum required bandwidth on stream i , $1 \leq i \leq K$.

Proof: Suppose the maximum difference of the aggregated bandwidth of the first i streams in the SFB scheme denotes d_i . From Lemma 1, we obtain

$$d_1 = 0$$

$$d_2 \leq \max(d_1, B_{\max}^2 - B_{\min}^2)$$

$$d_3 \leq \max(d_2, B_{\max}^3 - B_{\min}^3)$$

.

.

.

$$d_{K-2} \leq \max(d_{K-2}, B_{\max}^{K-1} - B_{\min}^{K-1})$$

$$d_K \leq \max(d_{K-1}, B_{\max}^K - B_{\min}^K).$$

By substituting the above equations, we can obtain $d_K \leq \max(B_{\max}^i - B_{\min}^i)$, where $1 \leq i \leq K$. \square

3.2 Viewers' Waiting Time

The viewer's waiting time comes from the access time of video segments on networks. To ensure continuous playout, the access time of a segment cannot be larger than its length. Thus, the maximum viewers' waiting time δ is equal to the length of a segment.

$$\delta = \frac{L}{N} \quad (6)$$

Suppose the video segments arranged by the SFB scheme are $X = \{X_j \mid X_j = S_{V[j]}\}$, where $1 \leq j \leq 2^K - 1$, and $V[j]$ is obtained from the algorithm in Fig. 2. The size of $X_{j > N}$ is zero. The segments numbered from 2^{i-1} to $2^i - 1$ are broadcasted on stream i sequentially, where $1 \leq i \leq K$. The segments transferred by the video server during $T_0 + (p-1)\delta$ to $T_0 + p\delta$ are $X_1, X_{2+(p-1)\delta}, \dots, X_{2^{K-1}+(p-1)\delta}, \dots, X_{2^{K-1}}$, where $1 \leq p \leq 2^{K-1}$. Suppose the size of X_j is x_j . The transferred data size during the period is

$$D_p = \sum_{q=1}^K x_{2^{q-1}+(p-1)\delta} 2^{q-1} \quad (7)$$

Therefore, the required bandwidth B_p is

$$B_p = \frac{1}{\delta} \sum_{q=1}^K x_{2^{q-1}+(p-1)\delta} 2^{q-1} \quad (8)$$

For a given bandwidth allocation B , the access time equals the transferred data size over the bandwidth. Suppose δ_p is the access time of the segments during $T_0 + (p-1)\delta$ to $T_0 + p\delta$. From equation (7), $\delta_p = \frac{1}{B} \sum_{q=1}^K x_{2^{q-1}+(p-1)\delta} 2^{q-1}$. Because the size of the segments is different, their access time varies with different time inter-

vals. Accordingly, the maximum waiting time δ_B at given bandwidth B equals the maximum δ_p , where $1 \leq p \leq 2^{K-1}$.

$$\delta_B = \max(\delta_p), \text{ where } \delta_p = \frac{1}{B} \sum_{q=1}^K x 2^{q-1+(p-1) \bmod 2^{q-1}}, \text{ and } 1 \leq p \leq 2^{K-1} \quad (9)$$

Figure 3 depicts the maximum bandwidth requirements for the movie, Jurassic Park III, using the FB scheme and the SFB scheme. The video is encoded by MPEG-2. Its length and size are 4800 seconds and 2.66 GBytes. For simplicity, we slightly change the FB scheme, which divides the video into equal-length segments and distributes the segments at constant bit rate, rather than at variable bit rate. This change reduces the maximum required bandwidth of the FB scheme. Even so, the figure shows that the SFB scheme still consumes less bandwidth than the FB scheme. In addition, with the increasing of segment length, the number of segments decreases so the number of the required streams (or the required bandwidth) also becomes small.

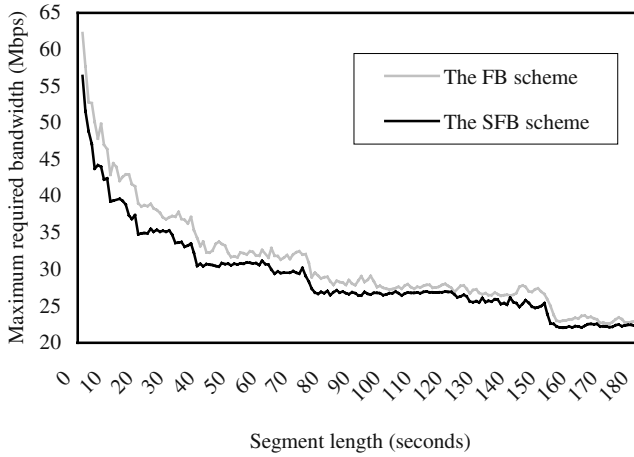


Fig. 3. The maximum required bandwidth versus segment length in the movie, Jurassic Park III

3.3 Buffer Requirements

The client needs to buffer portions of the playing video on disk because the arrival rate of the video data is larger than the consumption rate. In addition, the client merely buffers same video data once. Suppose the time that a client begins to receive video data is $T_0 + u\delta$. During $T_0 + u\delta + (p-1)\delta$ to $T_0 + u\delta + p\delta$, the segments that come from $C_{\lceil \log_2 p \rceil + 1}, C_{\lceil \log_2 p \rceil + 2}, \dots, C_K$ need to be buffered. Let

$$\begin{cases} I_p^u = \sum_{q=\lceil \log_2 p \rceil}^K X 2^{q-1+(u+p-1) \bmod 2^{q-1}}, & \text{where } 0 \leq u \leq 2^{K-1}-1 \text{ and } 1 \leq p \leq 2^{K-1} \\ I_p^u = 0, & \text{where } 0 \leq u \leq 2^{K-1}-1 \text{ and } 2^{K-1}+1 \leq p \end{cases} \quad (10)$$

represent the size of the increasing data that are written into the buffer by the client during this time interval. During the same interval, the client consumes previous received segments because the client cannot download and play a segment concurrently. Let

$$\begin{cases} O_1^u = 0 \\ O_p^u = S_{p-1}, & \text{where } 0 \leq u \leq 2^{K-1}-1, \text{ and } 2 \leq p \leq 2^K \end{cases} \quad (11)$$

represent the output size of the data that are read out from the buffer by the client during $T_0 + u\delta + (p-1)\delta$ to $T_0 + u\delta + p\delta$. Let Z_p^u represent the size of the required buffer during the same period. At $T_0 + u\delta + \delta$, all the data that come from C_1, C_2, \dots, C_K need to be buffered. Hence, we obtain

$$\begin{cases} Z_1^u = I_1^u \\ Z_p^u = Z_{p-1}^u + I_p^u - O_p^u, & \text{where } 0 \leq u \leq 2^{K-1}-1 \text{ and } 2 \leq p \leq 2^K-1 \end{cases} \quad (12)$$

During $T_0 + u\delta + (2^K-1)\delta$ to $T_0 + u\delta + 2^K\delta$, the client stops downloading the data, and begins consuming the last segment. There is no write requirement, and all the buffered data will be consumed during this interval. Hence, we obtain $Z_{2^K}^u = 0$.

According to equations (10), (11), and (12), we can calculate $\{Z_1^0, Z_2^0, \dots, Z_{2^{K-1}}^0, \dots, Z_1^{2^{K-1}-1}, \dots, Z_{2^{K-1}}^{2^{K-1}-1}\}$ for a fixed K . From equations (1) and (6), we can obtain $K = \left\lceil \log_2 \left(\frac{L}{\delta} + 1 \right) \right\rceil$; thus we can derive the relationship between the $\max \{Z_p^u \mid u = 0, \dots, 2^{\lceil \log_2 (\frac{L}{\delta} + 1) \rceil - 1} - 1; p = 1, \dots, 2^{\lceil \log_2 (\frac{L}{\delta} + 1) \rceil - 1}\}$ and the segment length δ . Figure 4 depicts the curve for the movie, Jurassic Park III. The SFB scheme performs quite well, and its maximum required buffer is far smaller than that of the FB scheme.

3.4 Disk Transfer Rate

The client must write the input video data into disk, as the data need to be buffered. When the client consumes the data, the client needs to read the data from disk. The

disk transfer (input/output) rate requirements are the sum of the read requirements and the write requirements.



Fig. 4. The maximum buffer requirements versus segment length in the movie, Jurassic Park III

From equation (10), the write requirements during $T_0 + u\delta + (p-1)\delta$ to $T_0 + u\delta + p\delta$ are

$$\begin{cases} W_p^u = \frac{1}{\delta} \sum_{q=\lceil \log_2 p \rceil}^K x 2^{q-1+(u+p-1) \bmod 2^{q-1}}, \text{ where } 0 \leq u \leq 2^{K-1}-1 \text{ and } 1 \leq p \leq 2^{K-1} \\ W_p^u = 0, \text{ where } 0 \leq u \leq 2^{K-1}-1 \text{ and } 2^{K-1}+1 \leq p \end{cases} \quad (13)$$

The read transfer rate is equal to the data consumption rate. Because the video is VBR-encoded, the rate varies with time. For simplicity, we merely consider the maximum consumption rate of each segment. Let b_{S_i} represent the rate of S_i . During $T_0 + u\delta$ to $T_0 + u\delta + \delta$, the read transfer rate is zero because the client cannot download and play the first segment concurrently. Let

$$\begin{cases} R_1^u = 0 \\ R_p^u = b_{S_{p-1}}, \text{ where } 0 \leq u \leq 2^{K-1}-1 \text{ and } 2 \leq p \leq 2^K \end{cases} \quad (14)$$

represent the maximum read transfer requirements during $T_0 + u\delta + (p-1)\delta$ to $T_0 + u\delta + p\delta$. Thus, the maximum disk transfer rate requirements are $\Phi_p^u = W_p^u + R_p^u$, where $0 \leq u \leq 2^{K-1}-1$ and $1 \leq p \leq 2^K$. Figure 5 depicts the requirements for the movie, Jurassic Park III. The SFB scheme requires smaller disk transfer rate than the FB scheme.



Fig. 5. The maximum disk transfer rate versus segment length in the movie, Jurassic Park III

4 Conclusions

The video broadcasting service is already popular on Internet. In this paper, we proposed a FB-based broadcasting scheme for VBR video services. The smooth fast broadcasting (SFB) scheme can smooth the bandwidth consumption. We derive its maximum difference of the required bandwidth. The viewers' waiting time, buffer requirements, and disk transfer rate are also analyzed mathematically. In addition, we use a VBR video to evaluate the SFB, and the results indicate that the scheme can offer better performance on the required bandwidth, buffer, and disk transfer rate than the FB scheme. For a video, the maximum difference of its required bandwidth is less or equal to $\max(B_{\max}^i - B_{\min}^i)$, where B_{\max}^i and B_{\min}^i represent the maximum and minimum required bandwidth on stream i . Future research could be directed toward finding a better scheme that can further alleviate the variety of bandwidth consumption. We also plan to propose new approaches to broadcasting live VBR videos.

Acknowledgement. The authors would like to thank the National Science Council of the Republic of China for financially supporting this research under Contract No. NSC 92-2213-E-008-004.

References

1. Almeroth, K.C., Ammar, M.H.: The use of multicast delivery to provide a scalable and interactive video-on-demand service. *IEEE Journal on Selected Areas in Communications*, Vol. 14, No. 5 (1996) 1110–1122
2. Dan, A., Sitaram, D., Shahabuddin, P.: Dynamic batching policies for an on-demand video server. *Multimedia Systems*, Vol. 4, No. 3 (1996) 112–121

3. Huang, L.-H.: Segment Loss Recovery on Hot-Video Broadcasting. Master thesis, National Central University (2002)
4. Juhn, L.-S., Tseng, L.-M.: Fast broadcasting for hot video access. The 4th International Workshop on Real-time Computing Systems and Applications (1997) 237–243
5. Juhn, L.-S., Tseng, L.-M.: Staircase data broadcasting and receiving scheme for hot video service. *IEEE Transactions on Consumer Electronics*, Vol. 43, No. 4 (1997) 1110–1117
6. Juhn, L.-S., Tseng, L.-M.: Harmonic broadcasting for video-on-demand service. *IEEE Transactions on Broadcasting*, Vol. 43, No. 3 (1997) 268–271
7. Juhn, L.-S., Tseng, L.-M.: Fast data broadcasting and receiving scheme for popular video services. *IEEE Transactions on Broadcasting*, Vol. 44, No. 1 (1998) 100–105
8. Juhn, L.-S., Tseng, L.-M.: Enhanced harmonic data broadcasting and receiving scheme for popular video service. *IEEE Transactions on Computer Electronics*, Vol. 44, No. 2 (1998) 343–346
9. Juhn, L.-S., Tseng, L.-M.: Adaptive fast data broadcasting scheme for video-on-demand services. *IEEE Transactions on Broadcasting*, Vol. 44, No. 2 (1998) 182–185
10. Li, F., Nikolaidis, I.: Trace-adaptive fragmentation for periodic broadcasting of VBR video. The 9th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV'99) (1999)
11. Ozden, B., Rastogi, R., Silberschatz, A.: On the design of a low cost video-on-demand storage system. *Multimedia Systems*, Vol. 4, No. 1 (1996) 40–54
12. Paris, J.-F., Carter, S. W., Long, D.D. E.: Efficient broadcasting protocols for video on demand. The 6th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (1998) 127–132
13. Paris, J.-F.: A simple low-bandwidth broadcasting protocol for video-on-demand. The International Conference on Computer Communications and Networks (1999) 118–123
14. Paris, J.-F.: A broadcasting protocol for compressed video. The Euromedia'99 Conference (1999) 78–84
15. Saporilla, D., Ross, K., Reisslein, M.: Periodic broadcasting with VBR-encoded video. *IEEE INFOCOM 1999* (1999) 464–471
16. Tseng, Y.-C., Yang, M.-H., Chang, C.-H.: A recursive frequency-splitting scheme for broadcasting hot videos in VOD service. *IEEE Transactions on Communications*, Vol. 50, No. 8 (2002) 1348–1355
17. Tseng, Y.-C., Yang, M.-H., Hsieh, C.-M., Liao, W.-H., Sheu, J.-P.: Data broadcasting and seamless channel transition for highly demanded videos. *IEEE Transactions on Communications*, Vol. 49, No. 5 (2001) 863–874
18. Viswanathan, S., Imielinski, T.: Pyramid Broadcasting for video on demand service. *IEEE Multimedia Computing and Networking Conference*, Vol. 2417 (1995) 66–77
19. Yang, H.-C., Yu, H.-F., Tseng, L.-M.: Adaptive Live Broadcasting for Highly-Demanded Videos. *Journal of Information Science and Engineering*, Vol. 19, No3 (2003)
20. Yang, Z.-Y., Juhn, L.-S., Tseng, L.-M.: On Optimal Broadcasting Scheme for Popular Video Service. *IEEE Transactions on Broadcasting*, Vol. 45, No. 3 (1999) 318–322
21. Yang, Z.-Y.: The Telepresentation System over Internet with Latecomers Support. Ph.D. Dissertation, Department of Computer Science and Information Engineering, National Central University, Taiwan (2000)

Dynamic Management of UDDI Registries in a Wireless Environment of Web Services

Z. Maamar¹, H. Yahyaoui², Q.H. Mahmoud³, and F. Akhter¹

¹ Zayed University, Dubai, U.A.E

{zakaria.maamar,fahim.akhter}@zu.ac.ae

² Laval University, Quebec, Canada

hamdi.yahyaoui@ift.ulaval.ca

³ University of Guelph, Guelph, Canada

qmahmoud@cis.uoguelph.ca

Abstract. This paper presents mechanisms for managing the content of several Universal Description, Discovery, and Integration (UDDI) registries. These mechanisms are deployed in a wireless environment of Web services. By content, it is meant the announcements of Web services that providers submit to an UDDI registry. Unlike other initiatives in Web services field that consider a single UDDI registry and a wired communication infrastructure, this paper is concerned with the following aspects: several UDDI registries are deployed, there is no wired communication infrastructure between the UDDI registries, and absence of a centralized component that coordinates the UDDI registries. The solution presented integrates users and software agents into what we call messenger. Initially, software agents reside in users' mobile devices and cache a description of the Web services that satisfy their users' needs. Each time a user is in the vicinity of an UDDI registry, her software agent interacts with that registry so the details stored on Web services are submitted.

Keywords: Management, Web service, UDDI, Wireless.

1 Introduction and Motivation

Web services are emerging as a major technology for achieving automated interactions between distributed and heterogeneous applications. A Web service is an accessible application that other applications and humans can discover and trigger [2]. Various technologies back the deployment of Web services such as WSDL, UDDI, and SOAP [3]. Unlike other research initiatives in the field of Web services that consider a single UDDI registry and assume a wired and stable communication infrastructure, we are concerned with the following aspects:

- Several UDDI registries are spread across different regions. A registry is aware of the presence of other registry peers but does not carry out any exchange of information on its content with these peers. The UDDI registries may belong to different institutions, have different usage policies, and pose

different requirements on acceptable announcements and retrieval requests of Web services.

- There is no wired communication infrastructure connecting the UDDI registries together. The existing infrastructure is of type wireless. Poor reliability and absence of coverage are among the features of this infrastructure.
- Absence of a centralized component that manages and coordinates the UDDI registries. On one hand, each registry is independent in defining the announcements of providers it accepts and the retrieval requests of users it satisfies. The definition of what to accept and what to satisfy is based on a set of *UDDI registry-defined policies*. On the other hand, each provider is independent in selecting the UDDI registries to whom it will submit its announcements of Web services. The selection of where to announce is also based on a set of *provider-defined policies*.

In a Web services running-scenario, an UDDI registry participates in two operations. The first operation is to receive the announcements of the description of the Web services (also called services in the rest of this paper) from providers. The second operation is to search the registry for the services that satisfy users' needs. Examples of needs are multiple such as hotel booking and car rental. However, since the announcements of services are submitted to distributed UDDI registries, this definitely leads into a different content among the registries. Therefore, it is important to develop mechanisms for supporting the exchange of content between distinct UDDI registries.

Targeting the dynamic management of UDDI registries has some similarities with the problem of information *replication*. An immediate solution to our UDDI registry-dynamic management is to flood the communication infrastructure with the new content of any UDDI registry that has seen a change. Changes in UDDI registries may become frequent as the number of Web services continues to grow. While the flooding fits the context of wired communication infrastructures, the lack of a reliable and permanent communication infrastructure, as it is with us, is a major obstacle to deploy this solution. Indeed, Karakasidis and Pitoura have observed that the traditional database approaches of collecting, caching, and indexing data of interest in monolithic contexts becomes obsolete in global computing contexts [6]. Thus, **another alternative is required for dynamically managing the UDDI registries**. In this paper, we discuss how mobile users will be the vehicle of supporting the content exchange between the UDDI registries. This support is done in a transparent way because of our use of *Software Agents* (SAs) [4].

In our initiative, each UDDI registry is associated with a structure referred to as *cluster* of Web services. Several clusters are made available across the wireless communication infrastructure so providers can connect to the most appropriate one using various criteria such as proximity and workload status of a cluster. However, for tracking requirements a provider cannot be connected to more than one cluster, i.e., a provider cannot announce in the UDDI registries of multiple clusters. The cluster in which a provider posts its services for the first time is called *master*. Interesting is the situation where providers have similar

services but respectively announces them in separate UDDI registries. Unless some appropriate exchange mechanisms exist, an UDDI registry would never be aware of the existence of similar services in other registries. Besides that, for a user wishing to satisfy her needs by triggering or composing Web services, she should be given the opportunity to consider all the existing services regardless of where they are announced.

A part of our solution to the dynamic management of UDDI registries relies on users who are on the move and have mobile devices. The other part of the solution relies on software agents. Indeed, we integrate users and software agents to constitute what we call *messengers*. While residing in the mobile device of a user, the agent caches a description of the list of Web services that are involved in the satisfaction of one of the user's needs. On behalf of providers, users announce services in various UDDI registries to be associated with clusters denoted by *slaves*. Because users have mobile devices, mobile support stations manage these devices in terms of identifying their physical location and handling their messages/calls. A mobile support station communicates with mobile users within its radio coverage area. This area is called *wireless cell*. When a user enters a new cell (i.e., under the coverage area of a new mobile support station) an exchange of information occurs between the agent of the user and the UDDI registry, which enables an update of this registry's content. We mention that a user does not have to visit all the clusters. Her association with a mobile support station depends on her *route* to various places such as work, mall, etc.

Section 2 presents the agentification process of the dynamic management of UDDI registries. Section 3 outlines our implementation work. Section 4 presents related work. Finally, Section 5 draws our conclusion and presents future work.

2 Agentification of a Wireless Environment of Web Services

2.1 Rationale of Software Agents

A SA is a piece of software that autonomously acts to carry out tasks on the users' behalf [4]. In agent-based applications, it is accepted that users only need to specify high-level goals instead of issuing explicit instructions, leaving the how and when decisions to their respective agent.

Besides the availability of several approaches and technologies for the deployment of Web services (e.g., SOAP, UDDI, Salutation), they are all tailored to the context of wired situations. In a similar context, all the computing resources are fixed and connected through a permanent and reliable communication infrastructure. The application of these approaches and technologies to the context of wireless situations is not straightforward. Indeed, major adjustments are required because of multiple obstacles such as potential disconnections of mobile devices and unrestricted mobility of persons. These obstacles highlight the suitability of SAs as potential candidates to handle them. First, a SA is autonomous. Thus, it can make decisions on the user's behalf while this one is disconnected.

Second, a SA can be mobile. Thus, it can move from one host to another. A continuous network connectivity is not needed [1]. Third, a SA is collaborative. Thus, it can work with other agents that identify for example the providers of Web services. Finally, a SA is reactive. Thus, it can monitor the events that occur in the user's environment so appropriate actions can be taken.

2.2 Agent-Based Deployment

Our agentification of the dynamic management of the UDDI registries has resulted into identifying three types of agents: *provider-agent*, *UDDI-agent*, and *user-agent*. Fig. 1 illustrates the agents according to the clusters of Web services, the UDDI registries, the mobile support stations, and the wireless communication infrastructure. Usually, the coverage areas (circles in Fig. 1) of the mobile support stations overlap. However, the overlapping is not represented in order to keep the figure clear. Because the clusters of Web services are wirelessly connected (dashed lines in Fig. 1), a continuous exchange of the content of the different UDDI registries cannot occur. Fig. 1 also features the notion of messenger - (user,user-agent). Users with their mobile devices are always associated with one mobile support station. When a user moves to a different place which is outside the coverage area of a mobile support station, a *handover* occurs between this station and the new mobile support station covering this place.

A provider-agent identifies a provider (i.e., a business) that intends to post its Web services in the UDDI registries of the multiple clusters. However, a provider is only authorized to connect to one cluster of type master. The services announced in the UDDI registry of a master cluster are labelled with the word *internal*. This labelling helps in (i) identifying the UDDI registry where the services have been announced for the first time, (ii) knowing the location of the providers to whom the services belong, (iii) denying all the operations of user-agents that aim at updating the description of the services, and (iv) naming the UDDI-agent that ensures that the parameters of the services (e.g., execution cost, execution time) are satisfied during their execution. Since a provider only announces its Web services in one UDDI registry, messengers take care of the UDDI registries of the remaining clusters of type slave (since the type of a cluster is seen from a provider perspective, thus, a cluster can be at the same time master and slave). For announcement purposes in slave clusters, the messengers follow certain policies as it is going to be explained below. The services posted in the UDDI registry of a slave cluster are labelled with the word *external*. This labelling helps in (i) indicating that a third entity (i.e., messenger) has announced the services, (ii) informing that the services can always be the object of changes, and (iii) stating that the UDDI-agent cannot guarantee the execution parameters of the services. Wired connections support the interactions between provider-agents and UDDI-agents (Fig. 1). These interactions are for announcing new services or for updating or withdrawing the services already-published.

Because users are heavily engaged in the announcement of external Web services, the agreement of the respective providers of these Web services is required. For *privacy* (e.g., a provider does not want to announce all its services in

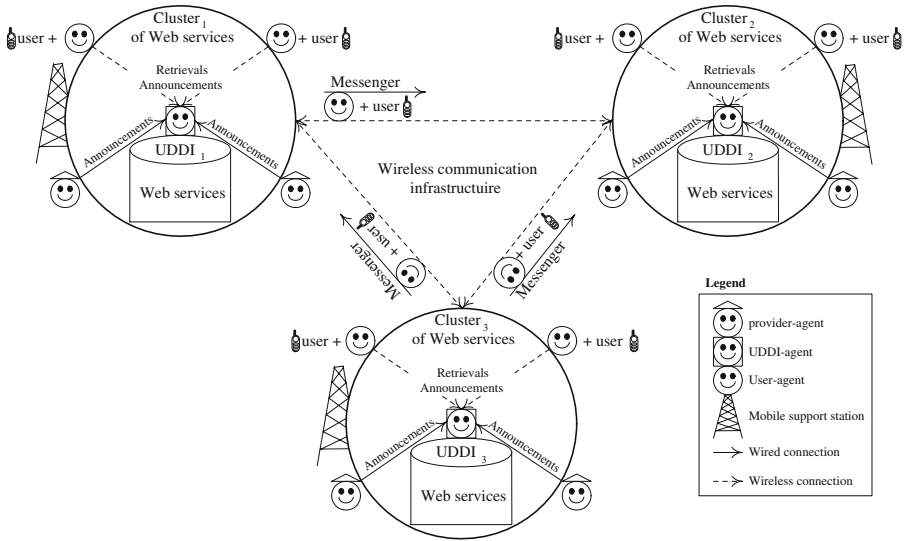


Fig. 1. Agentification of a wireless environment of Web services

a certain UDDI registry), *security* (e.g., a provider is afraid that its announcements of services will be altered in a certain UDDI registry), and *trustworthy* (e.g., a provider is not confident in the security mechanisms of a certain UDDI registry) reasons, a provider has to state in the description of a Web service that is submitted to an UDDI registry of a master cluster whether this service can be announced in the UDDI registry of a slave cluster. The statements of a provider are done through a set of policies that are attached to the announcements of services. Fig. 2 is a sample of a provider-defined trustworthy policy: provider-agent_{*i*} forbids to any user-agent the announcement of its Web services in UDDI-registry_{*j*}, if the trustworthy value between this provider and this UDDI registry is less than 0.5.

A user-agent resides in a mobile device (e.g., cell-phone, personal digital assistant) of a user constituting both a messenger when they move (Fig. 1). The main duty of a user-agent is to satisfy its user's needs (e.g., car rental, hotel booking) after it identifies and selects the relevant Web services with the help of an UDDI registry. To this purpose, the user-agent initiates interactions with the UDDI-agent of an UDDI registry. The selection of a specific UDDI registry depends on the current location of the user with regard to the mobile support station that is responsible of managing her mobile device. The composition of the services may also be required to satisfy certain users' needs (e.g., travel planning requiring flight reservation, hotel booking, user notification and probably car rental services) but this is beyond this paper's scope. Once the services are identified and triggered for execution, the user-agent caches in the user's mobile-device various information on these services such as the identifier of the services, the

```

if   trustworthy: (provider_agenti, uddi_registryj) < 0.5
then user_agent: not(announce(web_service, in(uddi_registryj)))

```

Fig. 2. Sample of a provider-defined policy

Note: The trustworthy value of a provider of Web services towards an UDDI registry is defined as follows. It is the number of times an UDDI registry has suggested with success external services of a provider (announced as internal services in other UDDI registries) to be involved in a composition process vs. the number of composition processes that were devised to satisfy users' needs (by success, it is meant triggered services). We recall that a UDDI-agent does not have a full control over the external services. For instance, it may occur that a service is announced in a slave UDDI-registry, but its provider has already decided to make the service unavailable for maintenance reasons.

UDDI registry with whom the user-agent has dealt with to obtain the services, and the providers of services and their location according to the master clusters. The information that a user-agent caches in a mobile device can be potentially announced in distinct UDDI registries of secondary clusters. This is done after verification of the authorization policies of the announcements (Fig. 2). A user-agent is not authorized to update the details of any service that is announced as internal in a master cluster.

By default, users are always attached to one cluster because of the mobile support station. When a user moves to a different place that is outside the coverage of this support station, her mobile device becomes under the management of a new mobile support station covering this place. This means that the user-agent can now start dealing with the UDDI registry of the new cluster of Web services. The user-agent keeps track of all the clusters it has visited, its last date of visit, and the kind of information it submitted to their respective UDDI registry. If the user-agent notes that the information it caches is beneficial to the UDDI-agent (what it was submitted *vs.* what it can submit now), a wireless communication is established between the two agents on a request-basis from the user-agent. The rationale of the communication is to transfer the information on the Web services to the UDDI registry so its content can be updated. The transfer has to obey to three categories of policies: provider-defined, UDDI registry-defined, and user-defined.

1. Provider-defined policies: the purpose of these policies is explained in the description of a provider-agent. A sample of a provider-defined policy is given in Fig. 2.
2. UDDI registry-defined policies: the purpose of these policies is to clarify for a UDDI-agent whether to accept announcements on external services and from which user-agent or UDDI registry. Fig. 3 is a sample of an UDDI registry-defined policy: UDDI-agent_i would not accept the announcements on Web services from any user-agent if this user-agent would have collected information on these services from UDDI-registry_j.
3. User-defined policies: the purpose of these policies is to define for a user-agent the UDDI registries of slave clusters where it can submit its announcements

```

if      source: (user_agent, web_service) = "uddi_registry"
then uddi_agenti: not(accept(web_service, from(user_agent)))

```

Fig. 3. Sample of an UDDI registry defined-policy

of external services. Fig. 4 is a sample of a user-defined policy: user-agent_i is authorized to announce its external services to UDDI-registry_j.

```

if  authorization: (user_agenti, uddi_registryj) = "yes"
then user_agenti: announce(web_service, in(uddi_registryj))

```

Fig. 4. Sample of a user-defined policy

A UDDI-agent runs on top of an UDDI registry. It interacts through wired connections with provider-agents for their announcements of services of type internal. Whereas, it interacts through wireless connections with user-agents for their retrieval requests of services and announcements of services of type external. The details on these interactions and their related policies have been explained above. With regard to managing the UDDI registries, it may happen that the announcements of Web services are not free-of-charges mainly when it comes to external services. In that case, a provider can specify a policy to avoid the posting of services in all the UDDI registries that request for charges. In Fig. 5, the policy states what follows: if provider-agent_i believes that its announcements of Web services through user-agents will be charged in UDDI-registry_j, then this provider-agent will give up declaring in this registry. Furthermore, it may happen that for the need of refreshing the content of an UDDI registry a UDDI-agent deletes the external services from its registry without notifying the respective provider. The opposite cannot apply to the internal services as the requests of deletion have to originate from their respective providers.

```

if      charges: (provider_agenti, uddi_registryj) = "yes"
then user_agent: not(announce(web_service, in(uddi_registryj)))

```

Fig. 5. Sample of a provider defined-policy

3 Implementation of the Messenger Approach

A proof-of-concept implementation of the messenger approach for the dynamic management of the UDDI registries is underway. The prototype uses

Sun's Java Web Services Developer Pack 1.2 (Java WSDP 1.2), which is an integrated toolkit for building, testing, and deploying Web Services (<http://java.sun.com/jwsdp>). Java WSDP comes with an implementation of an UDDI registry, which we integrate in our implementation. For the client side, we use Sun's J2ME Wireless Toolkit, which provides an implementation of the Java 2 Micro Edition (J2ME) (<http://java.sun.com/j2me>). We have chosen to deploy our prototype on handheld wireless devices such as Palm Pilots running PalmOS. This is mainly due to the fact that we cannot get easy access to wireless carrier equipments. The setup of this prototype environment is as follows:

- Three 802.11b wireless LANs are installed at different locations of the University of Guelph-Humber's campus.
- One UDDI registry is installed within the range of each wireless LAN.
- Users are equipped with PDAs (e.g., Palm Pilots). These PDAs, which are equipped with a wireless access card so they can wirelessly connect to the LANs, have the MIDP4Palm implementation installed on them. MIDP4Palm is a J2ME-based Java runtime environment for PalmOS devices.

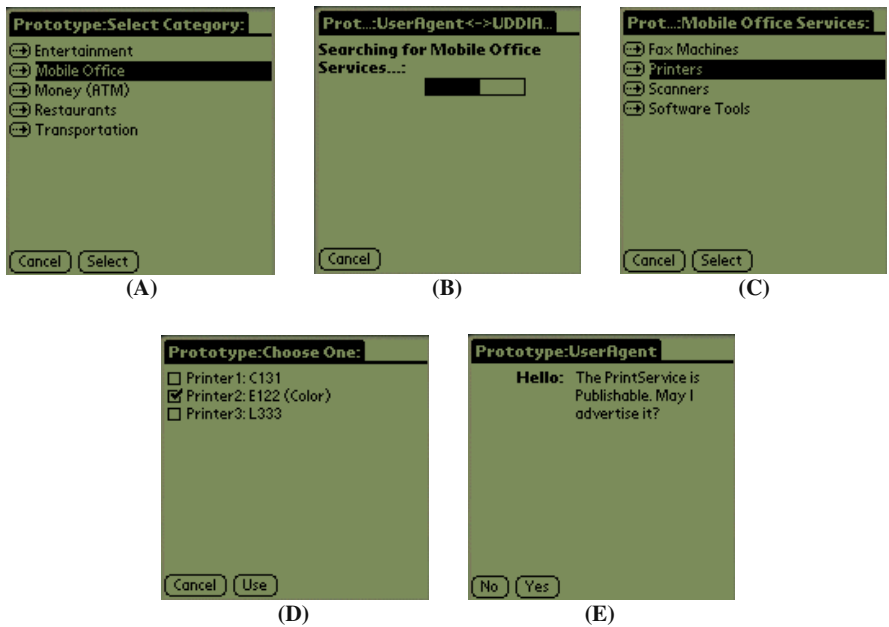


Fig. 6. Screen shots of the proof-of-concept of the messenger approach

As a first step, a number of Web services are developed and registered with different UDDI registries. The UDDI-agent is implemented as a Java-based server

that interacts with the UDDI registry as well as with the user-agent. The user-agent is implemented as a MIDlet network that can send messages across the network. Fig. 6 overviews the various screen shots of our proof-of-concept. Our running scenario is a user who needs a print service that would allow her to find the closest printer. Here are the steps that outline the messenger approach:

1. The user downloads and installs a MIDlet interface that allows her to interact with the UDDI registries and providers as well. Both are spread across different mobile support stations.
2. Once the MIDlet interface is installed (Fig. 6-A), it allows the user to search for a Web service within a specific category such as Mobile Office.
3. Afterwards, the user-agent communicates with the UDDI-agent within its wireless range in order to search for Web services in the Mobile Office category (Fig. 6-B).
4. If the search is successful, the UDDI-agent transfers a list of Web services to the user-agent so the user can select the appropriate service for example **PrintService** (Fig. 6-C).
5. After the user-agent stores a description on **PrintService**, a request triggering that service is submitted from the user-agent to the provider of this service so a search for printers within close proximity to the user is undertaken. The outcome of this search is a list of printers from which the user selects a printer (Fig. 6-D). The user may now remotely access the printer by clicking on the file to be sent out for printing.
6. Once the user has finished using **PrintService**, the user-agent checks if the UDDI registry and the respective provider of **PrintService** allow its posting in other UDDI registries. If it can be posted, the user-agent checks with the user whether she agrees too on publishing **PrintService** in other UDDI registries (Fig. 6-E).

4 Related Work

Scenarios where people on the move electronically interact with their surrounding environment have been reported in [5]. This backs our solution of getting users transparently involved in various operations, e.g. updating UDDI registries. Our work is at the crossing point of several research initiatives on Web services, UDDI registry, and wireless. While these concepts are being independently studied from each other (except for Web services and UDDI), we are aiming at their combination in the same framework. In what follows, we discuss the most related initiatives to ours.

In [8], MobiShare project provides a middleware system for offering ubiquitous connectivity to mobile devices. A mobile device is seen as a source of services, a requestor of services, or both. By service in MobiShare, it is meant the data that devices decide to publicly make available. Data availability depends on the status (on/off) and location (because of absence of coverage area) of a device. While in MobiShare the devices can be acting as providers of services, our devices have a different role. Indeed, they help in announcing the services of

providers in different locations (i.e., UDDI registries). Making announcements by users on behalf of providers is seen as a "favor" and not as a "commitment".

Service announcement and availability is another difference between our work and MobiShare. In MobiShare, each time a device moves from cell *A* (similar to a cluster) to cell *B*, the whole description of the services of the device moves also from cell *A* to cell *B*. A copy of this description remains in cell *A*, with a mention that the device is off-line (since it is outside the coverage area of cell *A*). Therefore, the description of the services is the same in cell *A* and cell *B*. In our work, the content of the UDDI registries after announcing services is different for various reasons: (i) each cluster has its own policy for accepting announcements from devices, (ii) each provider of services has the opportunity to decide where to announce its services, (iii) each user can decide whether to volunteer to be a messenger, and (iv) the number of users that transit by the coverage area of a cluster so they can make announcements.

In DBGlobe project, the aim is the development of data and metadata management techniques to deal with the challenge of global computing with a data-centric approach [6]. DBGlobe considers mobile entities as primary data stores and broadens the data management focus to address various issues such as mobility, autonomy, and scalability. To make data widely available, DBGlobe relies on chained hierarchies of directories. In case of an unsatisfied request at the level of a directory, the request is forwarded to a higher geographical authority. In our work, there are neither authorities nor hierarchies. All the clusters of Web services are at the same level. Indeed, clusters are independent in managing their respective UDDI registry.

5 Conclusion

In this paper, we presented our research initiative on the dynamic management of several UDDI registries deployed on top of a wireless environment of Web services. Our solution has relied on the fact that users are mobile as well as on the latest developments happening in the field of mobile devices (more storage capacity, more computing resources, and more advanced features). To manage the content of the UDDI registries, different types of policies have been put forward stating for example where to announce, what to announce, and what to accept. These policies have allowed us to consider several aspects in the announcement of Web services such as security, privacy, and trustworthiness. Because of the wireless communication infrastructure connecting the UDDI registries, a flooding-based solution has been discarded. Acting as messengers, users and their agents support the exchange of content between the UDDI registries.

Our current work is decomposed into several thrusts. Besides a test field of the proof-of-concept of the messenger approach, one of the thrusts is the security of the UDDI registries as agents of users may turn out to become malicious. Our security strategy takes advantage of our prior work on the security of Web services in the wireless world [7]. Another thrust consists of handling the situations where agents don't have enough time to complete the transfer of information from mobile devices to UDDI registries. Users may move outside the coverage areas that are associated with these UDDI registries.

References

1. P. Bellavista, A. Corradi, and C. Stefanelli. The Ubiquitous Provisioning of Internet Services to Portable Devices. *IEEE Pervasive Computing*, 1(3), July/September 2002.
2. B. Benatallah, Q. Z. Sheng, and M. Dumas. The Self-Serv Environment for Web Services Composition. *IEEE Internet Computing*, 7(1), January/February 2003.
3. F. Curbera, R. Khalaf, N. Mukhi, S. Tai, and S. Weerawarana. The Next Step in Web Services. *Communications of the ACM*, 46(10), October 2003.
4. N. Jennings, K. Sycara, and M. Wooldridge. A Roadmap of Agent Research and Development. *Autonomous Agents and Multi-Agent Systems*, Kluwer Academic Publishers, 1(1), 1998.
5. R. José, A. Moreira, and H. Rodrigues. The AROUND Architecture for Dynamic Location-Based Services. *Mobile Networks and Applications*, Kluwer Academic Publishers, 8(4), 2003.
6. A. Karakasidis and E. Pitoura. DBGlobe: a Data-Centric Approach to Global Computing. In *Proceedings of The 22nd International Conference on Distributed Computing Systems Workshops (ICDCSW'2002)*, Vienna, Austria, 2002.
7. Z. Maamar, H. Yahyaoui, W. Mansoor, and A. Bhati. Towards an Environment of Mobile Services: Architecture and Security. In *Proceedings of The 2003 International Conference on Information Systems and Engineering (ISE'2003)*, Montreal, Canada, 2003.
8. E. Valavanis, C. Ververidis, M. Vazirgiannis, G. C. Polyzos, and K. Norvag. Mo-biShare: Sharing Context-Dependent Data & Services from Mobile Sources. In *Proceedings of The 2003 IEEE/WIC International Conference on Web Intelligence (WI'2003)*, Halifax, Canada, 2003.

An Agent-Based Architecture for Service Discovery and Negotiation in Wireless Networks

Erich Bircher and Torsten Braun

University of Bern, Neubrückstrasse 10, 3012 Bern, Switzerland
braun@iam.unibe.ch

Abstract. This paper proposes a market-place based architecture, where users can detect wireless network services, negotiate with the identified service providers about price and service features, select the best service, and finally configure their devices according to the selected service. The architecture automates these different steps by the use of agents that represent the involved entities such as user, service provider and marketplace operator. The architecture has been implemented using FIPA-OS. Performance measurements show that procedures such as service discovery and service negotiation can be performed far below one second despite the significant overhead introduced by the FIPA-OS platform.

1 Introduction

Wireless local area networks (WLANs) became recently attractive for telecommunications operators to offer public wireless communication services. Despite the relatively high speeds in WLANs network resources are still scarce and are usually not offered for free or by flat rates. Such resources must be reserved and charged dependent on the resource usage. This requires some kind of a contract (also called service level agreement, SLA) between customer and service provider. On the other hand, the customer is interested to compare service offerings and select services with the best performance / price ratio. We propose to implement an environment based on agents and marketplaces where agents representing wireless service customers can detect and meet agents representing wireless network service providers, negotiate with them about the offered services, and reserve the resources for the agreed price. Marketplaces are well suited for realizing rendezvous places where services providers and customers can meet each other. Agents allow flexible negotiations among the involved entities and flexible configuration of end system software. After discussing related work in this area in Section 2, we will present our system architecture and implementation design in Section 3. This architecture is based on marketplaces that are used by service providers to register their wireless services in a certain geographical region and by users to discover appropriate service offerings. The architecture makes use of agents that communicate together behalf on the entities they represent such as user, marketplace and service providers. Section 4 discusses the performance results obtained with a FIPA-OS based implementation on Linux computers. Finally, Section 5 concludes the paper.

2 Related Work

The Foundation of Intelligent Physical Agents (FIPA) [1] is an organization for defining standards for multi agent systems. The key focus of FIPA is to specify

communication and inter-operability between agents in heterogeneous environments. In FIPA every agent is located on a platform. The different platforms are then linked together. A platform consists of three main parts: The agent management system for the agent life cycle management (management of platform, starting and deleting of agents, access control etc.), the directory facilitator, which provides yellow pages services, and an agent communication channel, which enables agents to communicate with each other. Each new agent has to register at the agent management system and at the directory facilitator. It can get information about other agents from the directory facilitator and can then contact the agents over the agent communication channel. These other agents can stay on the same platform or on another one, as long as the other platform is indirectly or directly linked together with this platform. The main part of FIPA is the definition of the communication between agents. Two agents are communicating with each other using a set of pre-defined protocols.

FIPA-OS [2] is an open source implementation of the FIPA standard. It is a component-based toolkit implemented in pure Java and supports most of the FIPA experimental specifications currently under development. Recently, a small version of FIPA-OS aimed at PDAs and smart mobile phones has been developed within the IST project Crumpet [3]. The IST Shuffle project uses an agent-based approach to control resources in UMTS networks. The project aims to create a novel architecture for efficient, scalable and robust real-time control of third generation mobile systems in the context of realistic business models of network providers, service providers and customers. This goal shall be reached with intelligent software agents complying to the FIPA standard.

3 System Architecture

3.1 Overall Architecture

The goal of this work is to realize a marketplace for temporary Internet access services via mobile devices. To realize a market it needs a buyer, a seller and a marketplace entity. These entities run on different computers and are connected over the Internet. The seller entity, representing the user, should support the user in service discovery and negotiation and manage the network connectivity of the portable device on which it runs. This means that it should change automatically to the access point where it has bought access time from the seller (representing an Internet service provider, ISP). The design should be adaptable to different connection technologies and to more complex and dynamic situations. This has to be considered especially for the software design. Based on all these requirements, we developed an overall system architecture that can be decomposed into three layers:

- The first layer is the network layer. It consists of the physical networks and computers such as the home network of the user or the hot spot location, where the user may go to and connect his portable device to the Internet, e.g. a wireless LAN hot spot. At such a hot spot there might be two ISPs offering wireless network access. The ISPs must have the wireless transmission environment and systems for IP address management such as DHCP servers. The ISP might also use extended services set IDs (ESSIDs), wired equivalent privacy (WEP), or password based schemes for access control. Another important part of the hot spot is the

been used as the programming language. This also requires a Java virtual machine running on each computer. The main entities of this layer are the user agent, located on the portable device of the user, the marketplace agent and the ISP agents, both likely installed at the hot spot location. Figure 2 shows a typical scenario where the user is in his home environment. He can use the services in the home environment such as email and file server access and can communicate with a correspondent node over the Internet. If the user plans to go to a place out of office, where he intends to connect to his home network, he can tell his user agent about his planned travel. The user agent is installed together with a FIPA-OS platform on the users portable device. The user agent contacts the corresponding marketplace agent, which returns a list of ISP agents. The user agent then negotiates a contract with one of these ISP agents. It gets the configuration data from the selected ISP after the negotiating a contract with the ISP agent. With this configuration data, it can later establish a network connection with the portable device at the desired location. If the users visits a hot spot without having a contract with an ISP, the user agent can contact an ISP at this hot spot over the reception access point in order to get a contract and the necessary network configuration data.

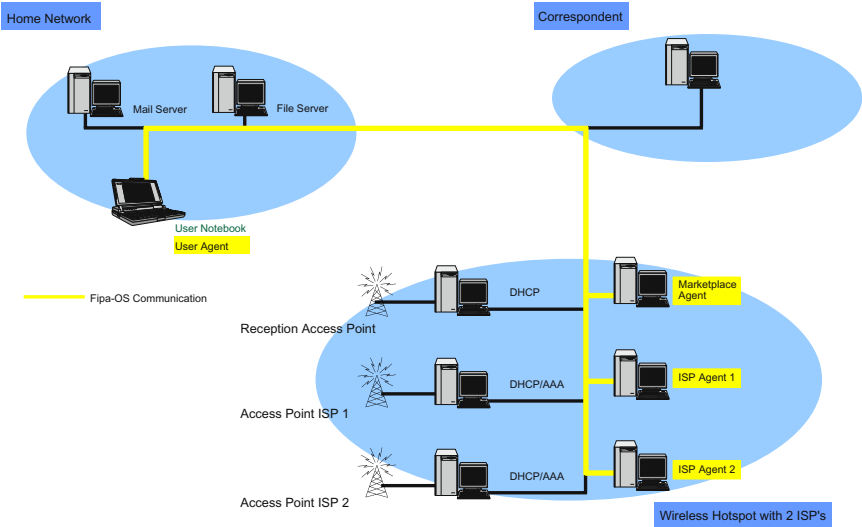


Fig. 2. Agents in the target scenario

3.2 Agent-Based Marketplace Design

For a simple market, only a seller and a buyer are needed. However, seller and buyer have to find each other. Marketplaces are places where seller and buyer can meet each other in order to negotiate and sell / buy services. We propose three entities (agents) for the buyer, the seller, and the marketplace.

The buyer entity represents the user in our case. It is therefore called the user agent. Its main task is to buy the desired services on behalf of the user. Additional tasks are communication with the user, negotiation with the ISPs and the appropriate configuration of the user's portable device. Therefore, the user agent has been split into three entities: the travel assistant (TA), the negotiation agent (NA) and the

configuration tool (CT). The main motivation behind this design choice is that the negotiation agent as a FIPA agent needs a running FIPA-OS platform. However, for many tasks of the user agent, the negotiation agent is not really needed. In this case, the FIPA-OS platform need not to be started but only the travel assistant. Moreover, the system should be extensible by new agents.

The travel assistant is the interface to the user and the controlling entity of the user agent. It supports several functions related to user travelling. In particular, it looks for connectivity during the time, when the user is out of office. The travel assistant communicates with the user over a graphical user interface (GUI) and delegates tasks to the negotiation agent and the configuration tool. It starts and stops the FIPA-OS Platform and the negotiation agent. The negotiation agent is an intelligent software agent that contacts the marketplace agents to get information about available locations and ISPs offering services there. It also negotiates a short-time service level agreement with eligible ISP agents. The configuration tool gets from the travel assistant all the negotiated contract information such as start time and duration of the service, the wireless network technology to be used and configuration data such as IP addresses, ESSIDs, WEP keys or user names and passwords. Later, it tries to establish the desired connections in time.

The ISP agent represents the seller, i.e. an ISP at a certain hot spot in our case. The main task of the ISP agent is to sell services on behalf of the ISP. It indicates its presence by subscribing to the according marketplace agent. If a user needs Internet access time at the hot spot, it performs SLA negotiations with the agent representing the interested user. In order to do this, it needs to have information about the ISP, the access point, available resources, and pricing schemes.

The role of the marketplace agent is to bring seller (ISP agent) and buyer (user agent) together. There are some different approaches how to design the marketplace entity. One approach is that the marketplace agent helps buyers (sellers) to find eligible sellers (buyers) and then let seller and buyer perform the deal by themselves. The approach is oriented to yellow pages and requires seller and buyer to be intelligent. The marketplace agent has only intermediation functionality. An other approach would be that the marketplace not only brings the buyer and seller together, but also performs the negotiation between them. This approach requires less intelligent sellers and bidders. They can just indicate their offers and bids to the marketplace. In cases, where the market should be controlled and where high trust is needed, such a centralized system would make much sense. The advantage of the decentralized approach is better flexibility and independence. If a seller wants to change his strategy it is far easier to change the corresponding agent instead of trying to indicate a new strategy to a marketplace agent. Thus, we have chosen the first approach for the realization of the marketplace. Figure 3 summarizes the agents and their relation between them.

A marketplace could be a huge directory covering all sellers in a large geographical region. On the other side, the markets can be organized in smaller units. We have chosen the latter approach: There is a marketplace for every access hot spot. The main motivation for that decision was that the marketplace agent can be located physically at the reception area. This allows users without contracts to contact this local

marketplace agent and the local ISP agents more easily. In addition, this distributed approach is much more scalable.

Another issue is the information exchange between the user agent and the marketplace agent that has to provide some information about the sellers. We aim to have a flexible marketplace agent that is able to handle different kinds of queries. Some user agents may only need the addresses of ISP agents that sell a certain product, others might desire more information and filter eligible ISPs by themselves. This approach allows the user agent to give some information about sellers and recommendations to the user. In particular, the marketplace agent gives some information about the registered ISPs such as the supported technology or acceptable payment methods to the user agents. The user agent can then decide about eligible ISPs or inform the user about indicated options that were not supported by the user. The main task of the marketplace agent representing a geographical hot spot location for Internet access is to inform interested parties about all the ISP offering services within its area. This requires that all these ISPs have to register with the marketplace agent. The marketplace gives then the information out to interested negotiation agents upon request.

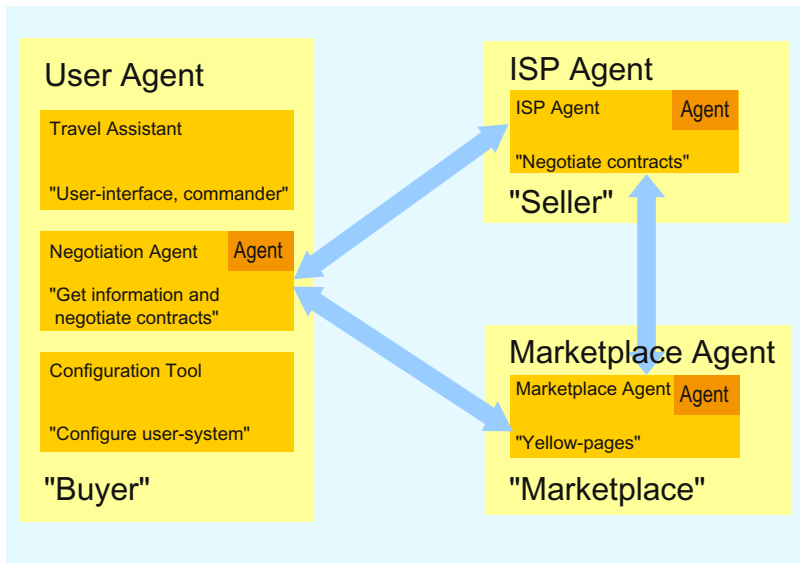


Fig. 3. Agent realizing a marketplace for wireless Internet services

3.3 Agent Interactions

In this subsection we describe in more detail how the agents interact with each other in order to fulfill the desired task. For agents it is important that they can understand each other. The usage of standard protocols, in particular FIPA protocols, is a significant step in order to solve this problem. The agents have to know which protocols to use in order to start a conversation (dialogue) with an other agent. Moreover, the agents have to understand the content they send to each other. This requires to clearly define the content to be exchanged. All participants in a conversation have to know the Java objects included in the messages. In our case,

three objects are exchanged between the agents and the objects need to be serialized for transmission.

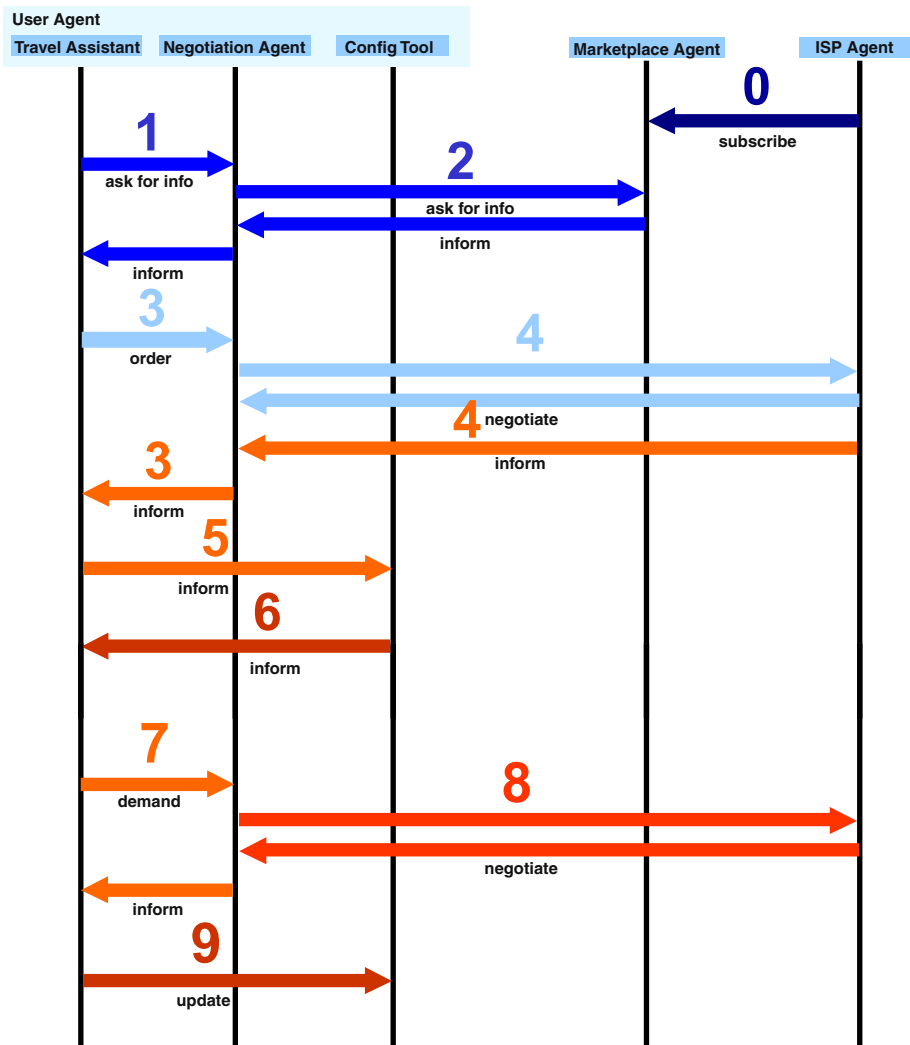


Fig. 4. Agent Interaction

Figure 4 shows the interactions among the various agents. First, an ISP agent has to register / subscribe at a marketplace agent delivering the ISP name, the access point technology and acceptable payment schemes (step 0). This step is supported by the FIPA-Subscribe-Protocol. Then, the travel assistant begins its operation by asking the negotiation agent to provide information about available marketplaces (1). This is done by internal communication based on method invocation between objects. The negotiation agent contacts the marketplace agent using the FIPA-Request-Protocol (2) and delivers the information about them (e.g. location and registered ISPs) back to the travel assistant. After that, the user finishes the interaction with the travel assistant

and gives a list with all desired connections and SLA information including target Quality-of-Service (QoS) values to the negotiation agent (3). The negotiation agent then contacts the potential ISPs and performs negotiations based on the SLA parameters proposed by the user (4). The protocol depends on the type of negotiation and can be a FIPA-Contract-Net-Protocol (Figure 5) or a FIPA-EnglishAuction-Protocol. The negotiation agent may now accept an offer from an ISP and receive configuration information after transmitting accounting information to the ISP agent. The negotiated SLA and the necessary configuration data are delivered to the travel assistant via the negotiation agent. In step 5, the travel assistant initiates the configuration of network and protocol parameters in order to be able to use the selected service to the configuration tool, which returns an acknowledgement (6). Steps 7-9 are performed in case of SLA modifications and similar to steps 3-5. For the negotiation in step 8 between the negotiation agent and the ISP agent the FIPA-Request-Protocol is used.

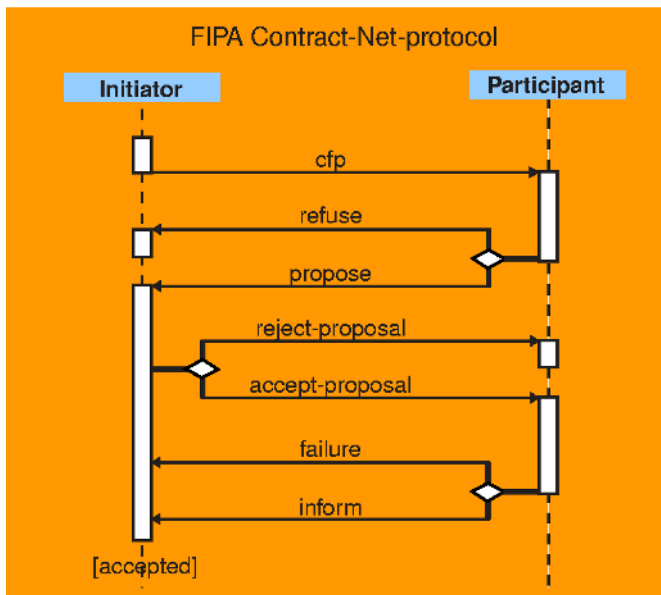


Fig. 5. FIPA-Contract-Net-Protocol

3.4 Market Issues

For price determination the following alternatives are possible: reverse-auctions, negotiations and fixed price. Auctions do not make much sense, because the delay for fixing the price would be too large. For stock markets, the offered service might not be homogeneous enough, since the service offerings need to be tailored to individual customer needs. Until now, we have implemented a fixed price approach, where the ISP agents return a fixed price dependent on bandwidth requirement and network connection time. At the user agent, a decision function according to [5] has been implemented in order to allow users to weight QoS parameters such as delay, bandwidth, packet loss etc. differently. The user can also specify preferences concerning price and contractual issues. For each parameter, the difference between

the actual and the ideal value is calculated and weighted. The offer with the lowest sum of weighted differences is preferred.

3.5 Security Issues

Agent-based negotiations implicitly introduce security risks. The negotiating entities should not only encrypt the exchanged information that might include sensitive data such as credit card numbers, but also authenticate each other. Confidential information exchange could be achieved using the Secure Socket Layer (SSL) for Java RMI. Authentication mechanisms could rely on public key infrastructures (PKI). The FIPA-OS version we used does not include any PKI support. However, this topic is currently being addressed by the research community [6].

3.6 Graphical User Interfaces

The different agents need also to provide a graphical user interface (GUI) in order to exchange information with a human who controls and configures the agent. The GUI of the ISP agent (Figure 6) allows to retrieve information from places and to register / deregister ISP agents at certain marketplaces. Moreover, it allows to compose service offerings including price information and network configuration data. The GUI for the marketplace agent displays the information about the registered ISP agents.

The screenshot shows the 'Blue' ISP Agent GUI. It is a windowed application with a title bar. The main area is divided into several sections:

- Agent ID:** Blue@localap
- ISP Name:** Blue
- Techniques:** 802.11b (dropdown menu)
- Accepted Payment Methods:**
 - ☒ Mastercard
 - ☒ VISA
 - ☐ American Express
 - ☒ Bank Account
- Marketplaces:**
 - Hauptbahnhof Zuerich
 - Seepromenade Zuerich
- registered at:** Airport Zuerich
- Buttons:** get Market info, register >, < deregister
- Proposal:**
 - Bandwidth (kbps): 2048
 - Latency (ms): 3
 - Jitter (ms): 3
 - Packet Loss (%): 2.0
 - Availability (%): 99.0
 - Price (\$/min): 0.1
 - Credits (min): 20
 - Payback 100% (h): 96
 - Payback 50% (h): 48
- Configuration-Data for Customers:**
 - Access Point Name: Cell_1
 - Key: A56BC6F87D
 - IP: (empty)
 - User Name: (empty)
 - Password: (empty)
- Status:** Accepted... (text field)
- Shutdown:** (button)

Fig. 6. ISP Agent GUI

The user agent GUI (Figure 7) allows interaction with the user. First, it provides the available host spot locations to the user. Moreover, the user can define preference profiles for different applications, e.g. file transfer, email, video conferencing, web browsing, in order to define bandwidth and QoS requirements for each type of application. The default values for the selected services are used for SLA negotiation unless the user overwrites these values. Additional parameters such as start and stop

time of the network service have to be defined prior to SLA negotiation. The user has also to define the network interfaces supported by its computer and the payment schemes he is willing / able to use.

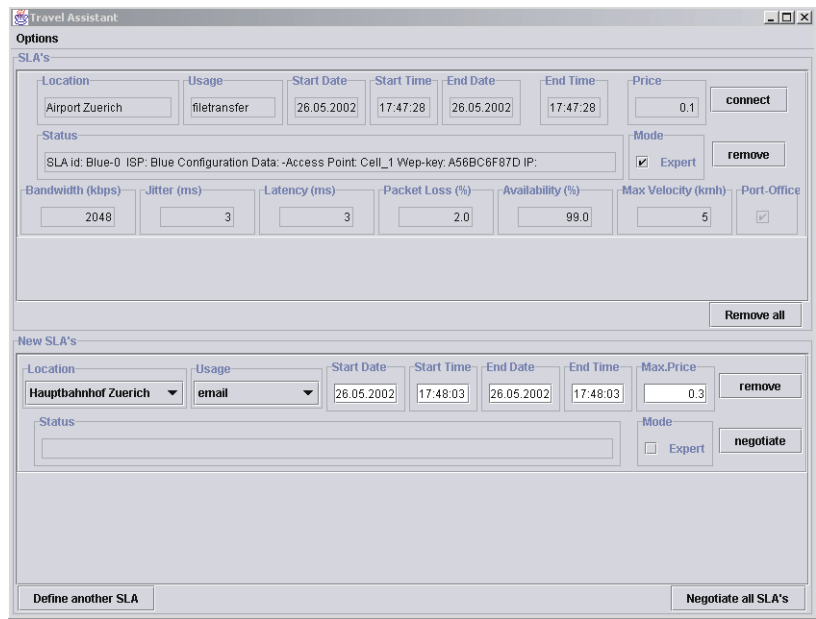


Fig. 7. User Agent GUI

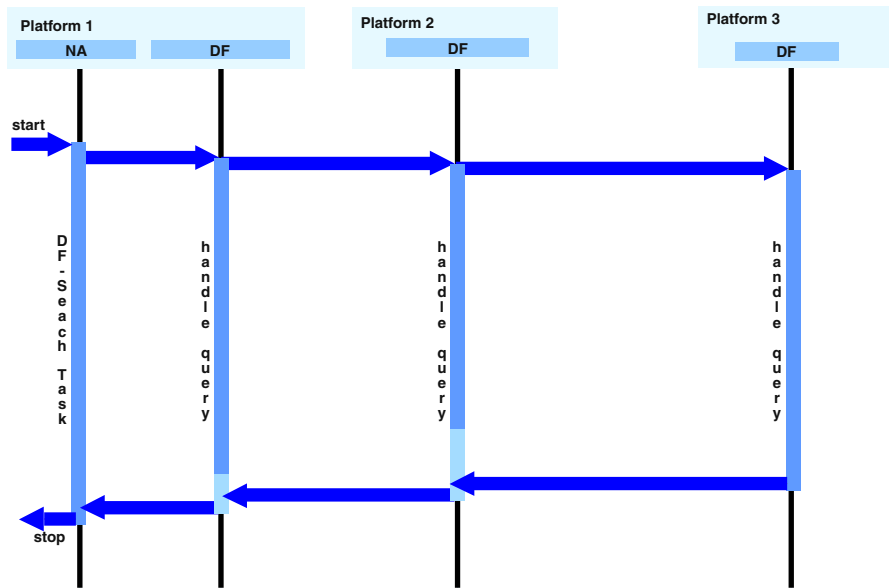


Fig. 8. Directory Facilitator Search across 3 Platforms

4 Performance Evaluation

For the performance evaluation Linux based agent implementations and IEEE 802.11 WLANs have been used.

4.1 Directory Facilitator

In a first test, we evaluated the performance for finding other registered agents in a cluster of computers. This is supported by the directory facilitator functionality which allows to search for specific agents on a platform and to forward such requests across different platforms that are linked together. The directory facilitator search is initiated by the negotiation agent and forwarded across a chain of 1-3 platforms (Figure 8). Platforms 1 and 2 were running on 333 MHz computers, while platform 3 was running on a 525 MHz computer. For a single platform, the directory facilitator search took approximately 900 ms in average, while the time increased to 3 s when introducing a second platform. However, introducing a third platform did not increase the search time. Obviously, platform 2 forwards the requests further and the search operations are performed in parallel at platforms 2 and 3.

4.2 Negotiation between Negotiation Agent and ISP Agent

While the directory facilitator search is only performed at system initialization, other conversations among the agents are more time-critical. First, we looked at one of the most complex conversations between the negotiation and the ISP agent: the negotiation that is based on the FIPA-Contract-Net-Protocol. When negotiation agent and ISP agent are running on a single computer (333 MHz), the conversation takes 605 ms in average. Distributing these two agents to two computers (333 MHz) increases the negotiation time slightly to 625 ms. Running the negotiation agent on a 333 MHz computer and the ISP agent on a 525 MHz computer reduces the negotiation time to 496 ms in average. These results show that ISP agent processing is rather costly and distributing agents to different computers can result in accelerating the negotiations.

The next series of tests have been performed with two ISP agents. The negotiation agent and one ISP agent were running on a 333 MHz computer each, one ISP agent was running on the 525 MHz computer. The negotiation time increased to 926 ms in average.

Finally, we tested the maximum capacity of an ISP agent by permanently issuing negotiations from a negotiation agent to an ISP agent. If negotiation agent and ISP agent were running on the same computer (333 MHz), 3.1 negotiations per second could be achieved. When distributing negotiation agent and ISP agent to two computers (both 333 MHz), the number increased to 3.8 negotiations per second. Using five negotiation agents communication to a single ISP agent increased the number even to 5.4 negotiations per second.

5 Conclusions

The paper described the realization of a marketplace for trading wireless network services based on agent technology. The FIPA technology has proven to be a viable candidate for the design and implementation of a marketplace where users and service

providers can meet. In particular, several components such as FIPA protocols have been very helpful. The preliminary performance results indicate that at least in scenarios with a few users in a hot spot area, the marketplace implementation achieves reasonable performance even on legacy computer systems. The concept also has significant potential for including more intelligence into the agents. Nevertheless, the performance of platforms need to be improved significantly and more systems such as PDAs and mobile phones need to be supported by such agent platforms.

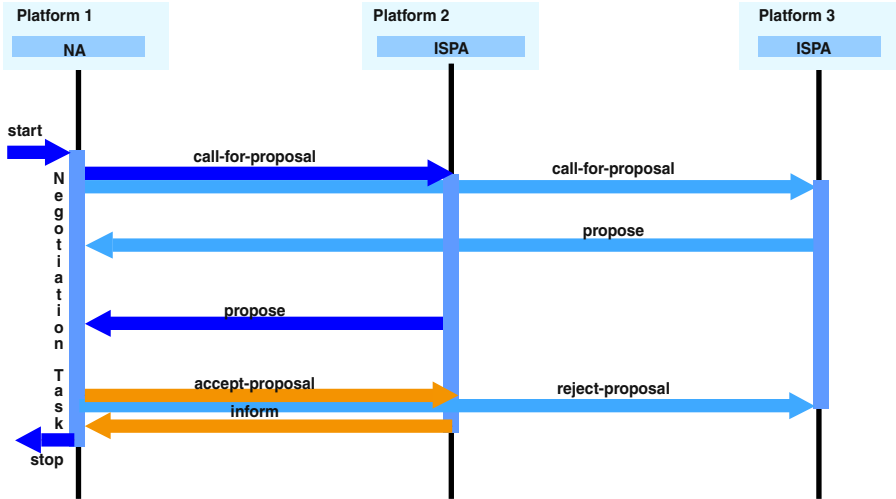


Fig. 9. Negotiations between 2 ISPs on 3 Platforms

Acknowledgements. The authors would like to thank Swisscom AG for providing infrastructure support to implement the architecture presented in this paper.

References

1. P. D. O'Brien, R.C. Nicol: FIPA-towards a Standard for Software Agents, BT Technology Journal, Vol.16, No.3, July 1998
2. S. Poslad, P. Buckle, and R. Hadingham. The FIPA-OS agent platform: Open Source for Open Standards, Firth International Conference and Exhibition on the Practical Application of Intelligent Agents and Multi-Agents, pp. 355-368, 2000.
3. S. Poslad, H. Laamanen, R. Malaka, A. Nick, P. Buckle and A. Zipf: CRUMPET: Creation of User-friendly Mobile Services Personalised for Tourism, Second International Conference on 3G Mobile Communication Technologies, March 26-29, 2001, London
4. Marc Danzeisen and Torsten Braun: Secure Mobile IP Communication, Workshop on Wireless Local Networks at the 26th Annual IEEE Conference on Local Computer Networks (LCN'2001), Tampa, USA, November, 15-16, 2001
5. J. Morris, P. Ree, P. Maes: Sardine: Dynamic Seller Strategies in an Auction Marketplace, ACM Electronic Commerce 2000, October 17-20, 2000, Minneapolis, Minnesota
6. Yuh-Jong Hu: Some Thoughts on Agent Trust and Delegation, Fifth International Conference on Autonomous Agents, May 2001

Author Index

- Akhter, F. 284
Aldunate, R. 48
Alonso, J. 143
Altman, E. 87
Atienza, D. 26
Azouzi, R. El 87

Babanskaja, I. 165
Baras, J.S. 132, 236
Barman, D. 87
Bettahar, H. 62, 260
Bircher, E. 295
Bottazzi, D. 38
Bouabdallah, A. 62, 260
Braun, T. 295

Catthoor, F. 26
Challall, Y. 260
Chen, Y.-M. 272
Corradi, A. 38

Dietterle, D. 165
Dombrowski, K. 165
Dunkels, A. 143

Eberhardt, R. 13
Ebrahimirad, H. 99

Han, Y.-H. 74
Hollick, M. 201
Hwang, C.-S. 74
Hwang, S.-H. 74

Iannello, G. 153

Jalili-Kharaajoo, M. 109

Karir, M. 132
Kellil, M. 62
Kim, K. 225
Koo, I. 225
Korhonen, J. 248
Kraemer, R. 165
Kuo, C.-Y. 272

Lach, H.-Y. 62
Lin, C.-H. 1
Liu, C.-Y. 1
Liu, Q. 177

Lu, Q. 177

Maamar, Z. 284
Mahmoud, Q.H. 284
Maihöfer, C. 13
Mamagkakis, S. 26
Matsuda, T. 120
Matsushita, Y. 120
Matta, I. 87
McGee, J. 132
Mendias, J.M. 26
Min, S.-G. 74
Montanari, R. 38
Mpartzas, A. 26

Noubir, G. 186
Nussbaum, M. 48

Pena-Mora, F. 48
Pescapè, A. 153
Pouiklis, G. 26

Ritter, H. 143
Romdhani, I. 62

Saha, A. 213
Schiller, J. 143
Schmitt, J.B. 201
Schoch, E. 13
Seipl, C. 201
Soudris, D. 26
Srinivasan, R. 236
Steinmetz, R. 201

Thanailakis, A. 26
Tseng, L.-M. 272

Ventre, G. 153
Voigt, T. 143
Vollero, L. 153

Yahyaoui, H. 284
Yamamoto, M. 120
Yang, H.-C. 272
Yazdanpanah, M.J. 99
Yu, H.-F. 272

Zander, J. 225